

**MAKAHIKI AND SGSEAM: A SERIOUS GAME FRAMEWORK FOR  
SUSTAINABILITY AND STAKEHOLDER EXPERIENCE  
ASSESSMENT METHOD**

**A DISSERTATION SUBMITTED TO THE  
GRADUATE DIVISION OF THE  
UNIVERSITY OF HAWAI'I AT MĀNOA  
IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF**

**DOCTOR OF PHILOSOPHY**

**IN**

**COMPUTER SCIENCE**

**MAY 2015**

**By**

**Yongwen Xu**

**Dissertation Committee:**

**Philip M. Johnson, Chairperson  
David Chin  
Scott Robertson  
Lipyeow Lim  
Daniel Port**

Copyright 2015 by

Yongwen Xu



To my wife

Wen and our son

Christopher: Thanks for  
all the love, support, and pa-  
tience you have shown me while

I worked on my Ph.D. It  
is most deeply ap-  
preciated.



## **ACKNOWLEDGMENTS**

This research project could not have been completed without the help of a number of people. My deepest thanks go to Prof. Philip Johnson, who has been my most respectful advisor and mentor for my four fruitful years in the Collaborative Software Development Lab.

I'd also like to thank past and present members of CSDL in chronological order: Robert Brewer, George Lee, Michelle Katchuck, Carleton Moore and Jordan Takayama. Special thanks go to George Lee, who developed the initial version of Makahiki. The Makahiki research project would not have happened without your long hours and hard work.

I would like to thank my committee members Prof. David Chin, Prof. Scott Robertson, Prof. Lipyeow Lim, and Prof. Daniel Port for taking the time to read and evaluate this dissertation.

This research is supported in part by grant IIS-1017126 from the National Science Foundation and the funding from the Center for Renewable Energy and Island Sustainability (REIS).

## ABSTRACT

Sustainability education and conservation have become an international imperative due to the rising cost of energy, increasing scarcity of natural resource and irresponsible environmental practices. Over the past decade, running energy and water challenges is the focal point for sustainability efforts at both university and industry campuses. Designers of such challenges typically have three choices for information technology: (a) build their own custom in-house solution; (b) out-source to a commercial provider; or (c) use a minimal tech solution such as a web page and manual posting of data and results.

None of these choices are ideal: the custom in-house solution requires sophisticated design and implementation skills; out-sourcing can be financially expensive and impedes evolution; and the minimal tech solution does not fully leverage the possibilities of advanced information technology.

To provide a better alternative to these three choices, I have led an effort over the past years to design and implement an open source serious game framework for sustainability called Makahiki. Makahiki implements an extensible framework with a variety of common services for developing sustainability games including authentication; game mechanics such as leaderboards, points, and badges; a variety of built-in games and content focused in sustainability, a responsive user interface, cloud-based deployment, and the ability to customize to the needs of individual organizations.

Makahiki lowers the overhead to those who would build a custom in-house solution by providing pre-built components. It can lower the financial cost to those who would out-source by providing an open source alternative. Finally, it provides an opportunity for those who would choose a minimal tech solution to instead provide more sophisticated information technology.

To provide initial evidence regarding the ability of the Makahiki framework to support sustainability games in different environments, we ran seven challenges at four organizations: The University of Hawaii at Manoa, Hawaii Pacific University, the East-West Center and Holy Nativity School. While these experiences provided anecdotal evidence for the usefulness of Makahiki, we realized that a more rigorous evaluation of the framework would yield better quality insight into its current quality and requirements for future enhancement.

Upon review of the literature, we found little research or experience with formal serious game framework assessment. To address this, I have embarked on research to design an assessment mechanism for serious game frameworks, called Serious Game Stakeholder Experience Assessment Method (SGSEAM). SGSEAM is designed to provide detailed insight into the strengths and weaknesses of a serious game framework through a stakeholder perspective based approach. In my research, I applied SGSEAM to Makahiki in order to gain better insight into its strengths and weaknesses as a serious game framework.

The contributions of my research thus includes: the Makahiki as the serious game framework for sustainability; the SGSEAM assessment method; the insights into creating and running a variety of real-world serious games for sustainability in different organizations; the insights into managing cloud based serious games; and the insights into serious game framework design and assessment method generated through application of SGSEAM to Makahiki. I hope this research will be of interest to researchers and practitioners across several disciplines: software engineering, game design, and sustainability research.

## TABLE OF CONTENTS

<b>Acknowledgments</b> . . . . .	<b>iv</b>
<b>Abstract</b> . . . . .	<b>v</b>
<b>List of Tables</b> . . . . .	<b>xviii</b>
<b>List of Figures</b> . . . . .	<b>xxii</b>
<b>1 Introduction</b> . . . . .	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Serious games and Gamification . . . . .	3
1.3 Game Design and Motivation . . . . .	4
1.4 Serious game Framework assessment . . . . .	5
1.5 Research Description . . . . .	6
1.5.1 Makahiki . . . . .	6
1.5.2 SGSEAM . . . . .	7
1.5.3 Evaluation . . . . .	8
1.6 Contribution . . . . .	10
1.6.1 Makahiki: A serious game framework for sustainability . . . . .	10
1.6.2 SGSEAM: a serious game framework assessment method . . . . .	10
1.6.3 Insights into the strengths and weaknesses of Makahiki . . . . .	10
1.6.4 Insights into creating and running a variety of real-world serious games . .	11
1.6.5 Insights into managing cloud based serious games . . . . .	11
1.6.6 Insights into the strengths and weaknesses of SGSEAM . . . . .	11
1.7 Outline . . . . .	12

<b>2 Related Work</b>	<b>14</b>
2.1 Sustainability and Education	14
2.2 Collegiate Sustainability Competitions	15
2.3 Serious Games and Gamification	19
2.4 Serious Games for Sustainability	25
2.5 Game Design and Motivation	30
2.6 Serious Game Frameworks	36
2.7 Serious Game and Framework Assessment	41
2.7.1 Game Metrics	45
2.8 Summary	47
<b>3 Makahiki Design</b>	<b>49</b>
3.1 Overview	49
3.2 System Description	50
3.2.1 System Admin Interface	51
3.2.2 Game Designer Admin Interface	52
3.2.3 Player Interface	56
3.2.3.1 Landing Page	56
3.2.3.2 Authentication Page	58
3.2.3.3 Home Page	59
3.2.3.4 Get Nutz Page	61
3.2.3.5 Go Low Page	64
3.2.3.6 Drop Down Page	65
3.2.3.7 News Page	66

3.2.3.8	Prizes Page . . . . .	67
3.2.3.9	Profile Page . . . . .	68
3.2.3.10	Help Page . . . . .	70
3.2.4	Game Manager Admin Interface . . . . .	71
3.2.5	Game Developer Admin Interface . . . . .	72
3.2.6	Game Analytics . . . . .	73
3.3	System Design . . . . .	74
3.3.1	Architecture . . . . .	74
3.3.2	A Library of Configurable Games and Mechanics . . . . .	77
3.3.2.1	Selecting a Game or Game Mechanics from the Library . . . . .	77
3.3.2.2	Energy and Water Game . . . . .	79
3.3.2.3	Smart Grid Game . . . . .	83
3.3.2.4	Top Score Game . . . . .	88
3.3.2.5	Raffle Game . . . . .	90
3.3.2.6	Participation Game . . . . .	93
3.3.2.7	Quest Game Mechanics . . . . .	94
3.3.2.8	Badge Game Mechanics . . . . .	96
3.3.2.9	Referral Bonuses Game Mechanics . . . . .	98
3.3.2.10	Configuring the games in a page . . . . .	100
3.3.3	Configurable resource . . . . .	101
3.3.4	Real-time Analytics . . . . .	103
3.3.5	Responsive mobile support . . . . .	105
3.3.6	Cloud deployment support . . . . .	106

3.3.7	Customization and Extension Development Support . . . . .	107
3.3.7.1	Theme Development . . . . .	107
3.3.7.2	New Game Widget Development Support . . . . .	108
3.4	Summary . . . . .	115
<b>4</b>	<b>SGSEAM Design . . . . .</b>	<b>117</b>
4.1	Overview . . . . .	117
4.2	Methodology . . . . .	119
4.3	Plan the Assessment . . . . .	120
4.3.1	Identify stakeholders . . . . .	120
4.3.2	Determine assessment approach . . . . .	121
4.3.2.1	Player Assessment . . . . .	122
4.3.2.1.1	Pre-Post effectiveness study . . . . .	122
4.3.2.1.2	Self-reported usability survey . . . . .	123
4.3.2.1.3	Engagement metrics . . . . .	123
4.3.2.2	System admin assessment . . . . .	125
4.3.2.2.1	Post-hoc admin interview . . . . .	125
4.3.2.2.2	In-lab system admin study . . . . .	126
4.3.2.3	Game designer assessment . . . . .	126
4.3.2.3.1	Post-hoc designer interview . . . . .	127
4.3.2.3.2	In-lab game design study . . . . .	127
4.3.2.3.3	Game design log data analysis . . . . .	128
4.3.2.4	Game manager assessment . . . . .	128
4.3.2.4.1	Post-hoc manager interview . . . . .	129

4.3.2.4.2	In-lab game management study . . . . .	129
4.3.2.4.3	Game management log data analysis . . . . .	129
4.3.2.5	Game developer assessment . . . . .	130
4.3.2.5.1	Post-hoc developer interview . . . . .	130
4.3.2.5.2	In-lab game development study . . . . .	130
4.3.3	Choose participants . . . . .	131
4.3.4	Create assessment schedule . . . . .	131
4.4	Gather Data . . . . .	131
4.4.1	Carry out the assessment . . . . .	132
4.4.2	Obtain log data . . . . .	132
4.5	Produce Assessment Report . . . . .	132
4.5.1	Analyze Data . . . . .	133
4.5.2	Determine strength and weakness . . . . .	133
4.5.3	Produce Report with Actionable Steps . . . . .	133
4.6	Summary . . . . .	134
<b>5</b>	<b>Makahiki and SGSEAM Evaluation . . . . .</b>	<b>135</b>
5.1	Real-world Makahiki Instances Case Studies . . . . .	135
5.2	Applying SGSEAM to Makahiki . . . . .	136
5.2.1	Step 1: Plan the Assessment . . . . .	136
5.2.2	Step 2: Gather Data . . . . .	137
5.2.2.1	Player Assessment . . . . .	138
5.2.2.2	System Admin Assessment . . . . .	139
5.2.2.3	Game Designer Assessment . . . . .	140

5.2.2.4	Game Manager Assessment . . . . .	141
5.2.2.5	Developer Assessment . . . . .	141
5.2.3	Step 3: Produce Assessment Report . . . . .	142
5.3	Consent from Human Research Study Subjects . . . . .	142
5.4	SGSEAM Assessment plan for Lucid BuildingOS . . . . .	142
5.5	Limitations . . . . .	143
5.6	Summary . . . . .	144
<b>6</b>	<b>Results . . . . .</b>	<b>146</b>
6.1	Real-World Case Studies . . . . .	146
6.1.1	Makahiki instances . . . . .	146
6.1.1.1	University of Hawaii at Manoa . . . . .	146
6.1.1.2	Hawaii Pacific University . . . . .	146
6.1.1.3	East-West Center . . . . .	147
6.1.1.4	Holy Nativity School . . . . .	148
6.1.2	Customization of the Makahiki Instances . . . . .	148
6.1.2.1	Challenge Customization . . . . .	148
6.1.2.2	Content Customization . . . . .	149
6.1.2.3	Branding Customization . . . . .	150
6.1.2.4	System Configuration . . . . .	155
6.1.3	Cloud Hosting Experiences . . . . .	156
6.1.3.1	Heroku Cloud Service: Dynos and Database . . . . .	157
6.1.3.2	Amazon S3 Cloud Service . . . . .	160
6.1.3.3	Memcache Cloud Service . . . . .	160

6.1.3.4	Cloud Monitoring Service . . . . .	161
6.1.4	Summary . . . . .	163
6.2	SGSEAM Assessment Results . . . . .	164
6.2.1	Makahiki Player Assessment . . . . .	165
6.2.1.1	Pre Post effectiveness study . . . . .	165
6.2.1.2	Self-reported effectiveness survey . . . . .	167
6.2.1.3	Self-reported usability survey . . . . .	171
6.2.1.4	Engagement metrics . . . . .	173
6.2.2	Makahiki System Admin Assessment . . . . .	175
6.2.2.1	In-lab installation study . . . . .	176
6.2.2.2	Post-hoc System Admin Interview . . . . .	179
6.2.3	Makahiki Game Designer Assessment . . . . .	181
6.2.3.1	In-lab game design study . . . . .	181
6.2.3.2	Post-hoc Game Designer Interview . . . . .	183
6.2.4	Makahiki Game Manager Assessment . . . . .	184
6.2.5	Makahiki Developer Assessment . . . . .	186
6.2.6	Assessment Report . . . . .	188
6.2.7	Summary . . . . .	189
6.3	Discussion . . . . .	189
<b>7</b>	<b>Conclusion . . . . .</b>	<b>193</b>
7.1	Research Summary . . . . .	193
7.2	Contributions . . . . .	195
7.2.1	Makahiki: A serious game framework for sustainability . . . . .	196

7.2.2	SGSEAM: A serious game framework assessment method . . . . .	196
7.2.3	Insights into the strengths and weaknesses of Makahiki . . . . .	197
7.2.4	Insights into creating and running a variety of real-world serious games . . . . .	197
7.2.5	Insights into managing cloud based serious games . . . . .	198
7.2.6	Insights into the strengths and weaknesses of SGSEAM . . . . .	199
7.3	Future Directions . . . . .	200
7.3.1	Applying SGSEAM to other serious game frameworks . . . . .	200
7.3.2	Community or Consortium to expand the content and game library . . . . .	200
7.3.3	Scale and expand Makahiki . . . . .	201
7.3.4	Use in other PaaS Cloud platforms . . . . .	201
7.3.5	Enhancement to Makahiki framework . . . . .	202
7.3.5.1	Real-time player awareness . . . . .	202
7.3.5.2	Deep Facebook integration . . . . .	203
7.3.5.3	Customization of game analytics page . . . . .	203
7.3.5.4	Action Library Management System . . . . .	203
<b>A</b>	<b>Publication List . . . . .</b>	<b>205</b>
A.1	Journal Paper . . . . .	205
A.2	Conference Papers . . . . .	205
A.3	Workshop . . . . .	206
A.4	Poster . . . . .	206
<b>B</b>	<b>SGSEAM Plan for Makahiki . . . . .</b>	<b>207</b>
B.1	Identify SGSEAM Stakeholders in Makahiki . . . . .	207
B.2	Determine SGSEAM Approaches for Makahiki . . . . .	207

B.2.1	Player Assessment Plan . . . . .	208
B.2.2	System Admin Assessment Plan . . . . .	210
B.2.3	Game Designer Assessment Plan . . . . .	212
B.2.4	Game Manager Assessment Plan . . . . .	213
B.2.5	Developer Assessment Plan . . . . .	213
B.3	Choose Assessment Participants . . . . .	214
B.4	Create Assessment Schedule . . . . .	215
<b>C</b>	<b>SGSEAM Data Repository for Makahiki . . . . .</b>	<b>216</b>
C.1	Player Assessment Data . . . . .	216
C.2	System Admin Assessment Data . . . . .	217
C.3	Game Designer Assessment Data . . . . .	217
C.4	Game Manager Assessment Data . . . . .	218
C.5	Developer Assessment Data . . . . .	218
<b>D</b>	<b>SGSEAM Improvement Action Report for Makahiki . . . . .</b>	<b>219</b>
D.1	Makahiki Player Assessment Report . . . . .	219
D.1.1	Player Assessment Results . . . . .	219
D.1.2	Strengths and Weaknesses from Player's Perspective . . . . .	220
D.1.3	Improvement Actions from Player's Perspective . . . . .	220
D.2	Makahiki System Admin Assessment Report . . . . .	221
D.2.1	System Admin Assessment Results . . . . .	221
D.2.2	Strengths and Weaknesses from System Admin's Perspective . . . . .	221
D.2.3	Improvement Actions from System Admin's Perspective . . . . .	222
D.3	Makahiki Game Designer Assessment Report . . . . .	222

D.3.1	Game Designer Assessment Results . . . . .	222
D.3.2	Strengths and Weaknesses from Game Designer’s Perspective . . . . .	223
D.3.3	Improvement Actions from Game Designer’s Perspective . . . . .	223
D.4	Makahiki Game Manager Assessment Report . . . . .	224
D.4.1	Game Manager Assessment Results . . . . .	224
D.4.2	Strengths and Weaknesses from Game Manager’s Perspective . . . . .	224
D.4.3	Improvement Actions from Game Manager’s Perspective . . . . .	224
D.5	Makahiki Developer Assessment Report . . . . .	225
D.5.1	Developer Assessment Results . . . . .	225
D.5.2	Strengths and Weaknesses from Developer’s Perspective . . . . .	225
D.5.3	Improvement Actions from Developer’s Perspective . . . . .	225
<b>E</b>	<b>SGSEAM Guide for Lucid Building OS and Building Dashboard . . . . .</b>	<b>227</b>
E.1	Overview . . . . .	227
E.2	Step 1: Plan the Assessment . . . . .	228
E.2.1	Identify Stakeholders . . . . .	228
E.2.2	Determine Assessment Approach . . . . .	229
E.2.2.1	Assess Player Stakeholder Experience . . . . .	230
E.2.2.2	Assess System Admin Stakeholder Experience . . . . .	232
E.2.2.3	Assess Game Designer Stakeholder Experience . . . . .	233
E.2.2.4	Assess Game Manager Stakeholder Experience . . . . .	234
E.2.2.5	Assess Game Developer Stakeholder Experience . . . . .	236
E.2.3	Choose Assessment Participants . . . . .	237
E.2.4	Create Assessment Schedule . . . . .	237

<b>E.3</b>	<b>Step 2: Gather Data . . . . .</b>	<b>238</b>
E.3.1	Carry Out Assessments . . . . .	238
E.3.2	Obtain log data . . . . .	238
<b>E.4</b>	<b>Step 3: Produce Assessment Report . . . . .</b>	<b>239</b>
E.4.1	Analyze Data . . . . .	239
E.4.2	Determine Strength and Weakness . . . . .	239
E.4.3	Produce Report with Actionable Steps . . . . .	239
<b>F</b>	<b>Questionnaire for SGSEAM In-Game Survey . . . . .</b>	<b>240</b>
F.1	UHM 2011 in-game survey questionnaire . . . . .	240
F.2	UHM 2012 in-game survey questionnaire . . . . .	244
F.3	UHM 2014 in-game survey questionnaire . . . . .	245
<b>G</b>	<b>Google Forms for SGSEAM In-lab Experiments . . . . .</b>	<b>248</b>
G.1	System admin Assessment . . . . .	248
G.1.1	Makahiki Local Installation Google Form . . . . .	248
G.1.2	Makahiki Heroku Installation Log . . . . .	256
G.2	Game designer Assessment . . . . .	261
G.2.1	Makahiki Configuration Log Google Form . . . . .	261

## LIST OF TABLES

2.1	University energy competitions . . . . .	18
2.2	Serious games for sustainability . . . . .	30
2.3	Game mechanics and use in Makahiki . . . . .	37
2.4	BuildingDashboard and Makahiki framework comparison . . . . .	39
2.5	Social Game Metrics . . . . .	46
3.1	Installation process . . . . .	51
3.2	Makahiki Internal Manager Modules . . . . .	76
3.3	Analytics Widgets on “Status” Page . . . . .	104
3.4	Makahiki built-in games and mechanics . . . . .	115
3.5	Configurability in Makahiki . . . . .	116
4.1	SGSEAM Stakeholders . . . . .	121
4.2	SGSEAM approaches . . . . .	122
4.3	Assessment schedule in the plan document . . . . .	132
4.4	SGSEAM Improvement Action Report Template . . . . .	134
5.1	Makahiki Serious Game Instances . . . . .	136
5.2	SGSEAM Assessment Plan Overview for Makahiki . . . . .	137
5.3	BuildingOS Assessment Approaches . . . . .	143
6.1	Challenge Configuration Differences . . . . .	149
6.2	Content Differences . . . . .	150

6.3	System Configuration Differences . . . . .	155
6.4	Hosting Configuration Differences . . . . .	156
6.5	Heroku Hosting Configuration . . . . .	157
6.6	Amazon S3 storage usage for Makahiki instances . . . . .	160
6.7	SGSEAM assessments for Makahiki . . . . .	164
6.8	Interests in sustainability prior and after the KC (2011 UHM, n=43) . . . . .	167
6.9	Interests in sustainability prior and after the KC (2012 UHM, n=44) . . . . .	168
6.10	Self-reported Perception of the Kukui Cup in 2011 UHM KC (n=43) . . . . .	169
6.11	Self-reported Behavior Changes in 2012 UHM KC (n=45) . . . . .	170
6.12	Self-reported Usability in 2011 UHM KC (n=43) . . . . .	171
6.13	Self-reported Usability in 2014 UHM KC (n=18) . . . . .	171
6.14	Engagement Metrics for 2011 and 2012 UHM Kukui Cup . . . . .	173
6.15	Average time (minutes) for local installation steps (n=8) . . . . .	176
6.16	Makahiki local installation issues reported (n=8) . . . . .	177
6.17	Average time (minutes) for Cloud installation steps (n=8) . . . . .	178
6.18	Makahiki cloud installation issues reported from blog posts (n=8) . . . . .	178
6.19	Installation Issues in HPU 2012 KC . . . . .	180
6.20	Installation Issues in HPU 2013 KC . . . . .	180
6.21	Average time (minutes) for design tasks (n=8) . . . . .	182
6.22	Makahiki Game Design Analysis, (n=8) . . . . .	183
6.23	Makahiki Game Design Experiences in 2012 HPU Kukui Cup . . . . .	184
6.24	Makahiki Game Design Experiences in 2012 EWC Kukui Cup . . . . .	184
6.25	Makahiki Game Managing Experiences in 2012 HPU and EWC Kukui Cup . . . . .	185

6.26	Makahiki Game Development Experience, (n=8) . . . . .	187
6.27	Makahiki Game Development Problems, (n=8) . . . . .	188
B.1	SGSEAM Stakeholders . . . . .	207
B.2	SGSEAM approaches . . . . .	208
B.3	SGSEAM Participants for Makahiki . . . . .	214
B.4	SGSEAM Assessment Schedule for Makahiki . . . . .	215
C.1	Player Assessment Data . . . . .	216
C.2	Sys admin Assessment Data . . . . .	217
C.3	Game Designer Assessment Data . . . . .	217
C.4	Game Manager Assessment Data . . . . .	218
C.5	Developer Assessment Data . . . . .	218
D.1	SGSEAM Improvement Action Report from Player's Perspective . . . . .	220
D.2	SGSEAM Improvement Action Report from System Admin's Perspective . . . . .	222
D.3	SGSEAM Improvement Action Report from Game Designer's Perspective . . . . .	223
D.4	SGSEAM Improvement Action Report from Game Manager's Perspective . . . . .	225
D.5	SGSEAM Improvement Action Report from Developer's Perspective . . . . .	226
E.1	Applying SGSEAM to a framework . . . . .	228
E.2	BuildingOS Stakeholders . . . . .	229
E.3	BuildingOS Assessment Approaches . . . . .	230
E.4	Player Assessment . . . . .	231
E.5	System Admin Assessment . . . . .	233

E.6	Game Designer Assessment . . . . .	234
E.7	Game Manager Assessment . . . . .	235
E.8	Game Developer Assessment . . . . .	236
E.9	Choose Participants . . . . .	237
E.10	Assessment Schedule . . . . .	238

## LIST OF FIGURES

2.1	Hawaii Energy Sources [?]	15
2.2	Harvard Green Cup Competition for Energy	16
2.3	Oberlin College's Custom Designed Sustainability Competition [?]	17
2.4	University of British Columbia Energy Competition[?]	18
2.5	Serious Game Example: Foldit is solving a serious problem [?]	20
2.6	Serious Game Example: World Without Oil - Play it before you live it. [?]	21
2.7	Serious Game and Gamification (source: Deterding [?])	22
2.8	Gamification Example: Foursquare makes modern badges popular[?]	23
2.9	Gamification Example: Nike+ makes fitness run[?]	24
2.10	Gamification Example: RibbonHero Helps to Learn Office[?]	24
2.11	Gameful Design for Sustainability	26
2.12	Vermontivate Sustainability Game (source: Vermontivate [?])	26
2.13	RecycleBank - Gaming for Good	27
2.14	Social Energy Application Engaging Consumers (source: Opower [?])	28
2.15	Power House Game to Save Energy(source: Reeves [?])	29
2.16	EnerCities - simulation based sustainability game(source: EnerCities [?])	30
2.17	Ancient Games Shown in British Museum	31
2.18	The state of flow is achieved between anxiety and boredom [?]	32
2.19	Player Types	33
2.20	Basic game mechanics and elements	34

2.21 Gamification is about extrinsic rewards (source: Anderson [?]) . . . . .	35
2.22 Designing Player Journey (source: Kim [?]) . . . . .	36
2.23 Building Dashboard (source: Lucid [?]) . . . . .	38
2.24 Serious game lifecycle using a framework . . . . .	40
2.25 Stanford Energy Services Platform (source: Stanford [?]) . . . . .	41
2.26 Evaluation Framework for Serious games . . . . .	42
2.27 Technology Acceptance Model (TAM) [?] . . . . .	43
2.28 Existing assessment methods in a serious game lifecycle . . . . .	44
2.29 Existing assessment methods in a serious game lifecycle . . . . .	45
2.30 Player Metrics (source: Ducheneaut [?]) . . . . .	46
2.31 Social Game Metrics Example(source: Appdata.com [?]) . . . . .	47
2.32 Game Engagement Questionnaire (GEQ) items [?] . . . . .	48
 3.1 System Admin Settings Page . . . . .	53
3.2 Challenge Design Settings Page . . . . .	54
3.3 Smart Grid Game Designer Page . . . . .	55
3.4 Landing page . . . . .	56
3.5 Landing page for East West Center Kukui Cup . . . . .	57
3.6 Authentication page for University of Hawaii Kukui Cup . . . . .	58
3.7 Authentication page for East West Center Kukui Cup . . . . .	58
3.8 Home page . . . . .	59
3.9 Home page for Hawaii Pacific University Kukui Cup . . . . .	60
3.10 Get Nutz Page . . . . .	61
3.11 Another Get Nutz Page . . . . .	62

3.12 Action Page . . . . .	63
3.13 Go Low Page . . . . .	64
3.14 Drop Down Page . . . . .	65
3.15 News Page . . . . .	66
3.16 Prizes Page . . . . .	67
3.17 Profile Page . . . . .	68
3.18 Profile Page with the Google theme . . . . .	69
3.19 Profile Page with the Space theme . . . . .	69
3.20 Help Page . . . . .	70
3.21 Challenge Management Settings Page . . . . .	71
3.22 Developer Settings Page . . . . .	72
3.23 Status Page . . . . .	73
3.24 Architecture of Makahiki . . . . .	75
3.25 Internal Architecture of Makahiki . . . . .	76
3.26 Selecting a Game from Game Settings . . . . .	77
3.27 Change a Game's Setting . . . . .	78
3.28 Daily Energy Goal Game widget . . . . .	80
3.29 Power Meter widget . . . . .	81
3.30 Daily Energy Goal Game Calendar widget . . . . .	81
3.31 Daily Energy Goal Game Scoreboard . . . . .	82
3.32 Smart Grid Game widget . . . . .	83
3.33 Smart Grid Game Designer widget . . . . .	86
3.34 Grid Consistency Checker Widget . . . . .	87

3.35 Action Dependency Tree . . . . .	87
3.36 Top Score Game . . . . .	88
3.37 Specifying a TopScore Game Prize . . . . .	89
3.38 Raffle Game widget . . . . .	90
3.39 Specifying a Raffle Game Prize . . . . .	91
3.40 Participation Game Scoreboard . . . . .	93
3.41 Specifying a participation Game Settings . . . . .	94
3.42 Quest Game Mechanics . . . . .	95
3.43 Specifying a Quest Game Mechanics . . . . .	95
3.44 Badge widget . . . . .	97
3.45 Specifying a Badge Game Mechanics . . . . .	97
3.46 Referral Bonus Game Mechanics Widget . . . . .	98
3.47 Changing the Referral Bonus Game Mechanics Settings . . . . .	99
3.48 Configuring a Page . . . . .	100
3.49 Configurable resource . . . . .	101
3.50 Game analytic widgets: User Stats and Energy Goal Status . . . . .	103
3.51 Responsive design supports both desktop and mobile . . . . .	105
3.52 Heroku cloud deployment . . . . .	106
3.53 Change the Website Theme . . . . .	107
3.54 Makahiki Pre-Built Themes Defined in settings.py . . . . .	108
3.55 Google Theme defined in theme-google.less . . . . .	109
3.56 The Result of theme-google.less . . . . .	110
3.57 Makahiki Tool for Creating a New Widget Structure . . . . .	110

3.58 Supply function template provided in the base views.py . . . . .	111
3.59 Example of a completed views.py . . . . .	112
3.60 Generated base index.html . . . . .	112
3.61 Example of a completed index.html . . . . .	113
3.62 Adding the new widget to settings.py . . . . .	113
3.63 The Hello World widget added to the left column . . . . .	114
3.64 The Hello World widget displayed in the Profile page . . . . .	114
4.1 Applying SGSEAM to a serious game framework . . . . .	119
4.2 Player self-reported usability metrics questionnaires . . . . .	124
4.3 Player engagement metrics . . . . .	125
4.4 System admin interview questionnaires . . . . .	126
4.5 Game designer interview questionnaires . . . . .	127
4.6 Game manager interview questionnaires . . . . .	129
4.7 Game developer interview questionnaires . . . . .	130
6.1 UHM Kukui Cup Challenge Home Page . . . . .	147
6.2 HPU Kukui Cup Challenge Home Page . . . . .	147
6.3 EWC Kukui Cup Challenge Home Page . . . . .	148
6.4 HNS Kukui Cup Challenge Home Page . . . . .	149
6.5 UHM SmartGrid Game Layouts . . . . .	151
6.6 HPU SmartGrid Game Layouts . . . . .	152
6.7 EWC SmartGrid Game Layouts . . . . .	153
6.8 HNS SmartGrid Game Layouts . . . . .	154

6.9	Performance testing to determine the appropriate number of dynos . . . . .	158
6.10	Dynamic Heroku dyno allocation during the UHM 2012 instance . . . . .	159
6.11	Memory cache usage monitoring . . . . .	161
6.12	NewRelic application monitoring . . . . .	162
6.13	NewRelic database monitoring . . . . .	162
6.14	Literacy survey result from Brewer 's dissertation [?] . . . . .	166
6.15	Energy consumption result from Brewer's dissertation [?] . . . . .	166
6.16	Interests in sustainability prior and after the UHM 2011 KC . . . . .	167
6.17	Interests in sustainability prior and after the 2012 UHM KC . . . . .	168
6.18	Sample responses of reported increased interests in sustainability . . . . .	169
6.19	Sample responses of reported behavior changes . . . . .	170
6.20	Self-reported Usability Measurements in 2011 KC and 2014 KC . . . . .	172
6.21	Engagement Measurements during the 2011 and 2012 UHM Kukui Cup . . . . .	174
6.22	Average time for local installation steps (n=8) . . . . .	177
6.23	Average time for cloud installation steps (n=8) . . . . .	178
6.24	Average time for design tasks (n=8) . . . . .	182
6.25	Suggestion for Makahiki Game Development support . . . . .	188
7.1	Virtual Machines V.s. Docker [?] . . . . .	202
B.1	Makahiki Developer assessment Form . . . . .	211

# CHAPTER 1

## INTRODUCTION

In this dissertation, I describe the Makahiki research project, which explores the information technology to provide infrastructure to facilitate the development of serious games for sustainability. The research consists of an open source serious game framework for sustainability and a method for assessing a serious game framework based on the stakeholder experience. In this chapter I explain the motivation of the research, briefly describe the Makahiki system and the SGSEAM assessment method, finally summarize the results and contributions of this research.

### 1.1 Motivation

The rising cost, increasing scarcity, and environmental impact of fossil fuels as an energy source makes a transition to cleaner, renewable energy sources an international imperative. In Hawaii, the need for transition is especially acute, as the state leads the US both in the price of energy and reliance on fossil fuels as an energy source (over 90% from oil and coal in 2012[?]).

One barrier to this transition is the success that electrical utilities have had in making energy ubiquitous, reliable, and easy to access, thus enabling widespread ignorance in the general population about basic energy principles and trade-offs. Moving away from petroleum is a technological, political, and social paradigm shift, requiring citizens to think differently about energy policies, methods of generation, and their own consumption than they have in the past.

On the other hand, changing people's behavior with respect to energy holds significant promise in reducing energy use. Darby's survey of energy consumption research found that identical homes could differ in energy use by a factor of two or more [?]. Data from a military housing community on Oahu show energy usage for similar homes can differ by a factor of 4 [?].

Petersen et al. describe their experiences deploying a real-time feedback system in an Oberlin College dorm energy competition in 2005 that includes 22 dormitories over a 2-week period [?]. Web pages were used to provide feedback to students. They found a 32% reduction in electricity use across all dormitories.

Over the past decade, running energy and water challenges have become a focal point for sustainability efforts at university and industry campuses, to facilitate and incentivize energy and water reduction. In a 2011 survey [?], Hodge found that 163 universities and colleges held or planned to hold an energy competition during the 20102011 academic year. 40% of these organizations are holding a competition for the first time. Hodge found that the average reduction in electricity use during these competitions is 9%.

Designers of such challenges typically have three choices for information technology: (a) build their own custom in-house solution (as was done at Oberlin College in 2006 [?]); (b) out-source to a commercial provider (as was done at the University of British Columbia in 2011 [?]); or (c) use a minimal tech solution such as a web page and manual posting of data and results (as was done at Harvard University in 2012 [?]).

None of these choices are ideal: the custom in-house solution requires sophisticated design and implementation skills; out-sourcing can be financially expensive and impedes evolution; and the minimal tech solution does not fully leverage the possibilities of advanced information technology.

To provide a better alternative to these three choices, Makahiki is designed as an open source serious game framework for sustainability with the goal of providing the infrastructure to enable different organizations to easily create and manage sustainability related serious games including this kind of sustainability challenges.

The closest technology to Makahiki of which we are aware is the Building Dashboard [?] that supports the Campus Conservation Nationals [?], which has been used by over 100 schools nationwide in 2014 to implement an energy and water reduction competition, including the above mentioned University of British Columbia. The Building Dashboard enables viewing, comparing and sharing building energy and water use information on the web in a compelling visual interface. But it focused on providing energy usage as passive information. There is little interaction between participants and the system. Unlike Makahiki, the Building Dashboard system does not support game mechanics, education, or synergy between real and virtual world environments. In addition, it is not open source. The cost of the system creates a barrier for wider adoptions.

## 1.2 Serious games and Gamification

Another emergent issue is the explosive spread of game techniques, not only in its traditional form of entertainment, but across the entire cultural spectrum. Games have been shown with great potential as successful interactive media that provide engaging interfaces in various serious contexts [?, ?]. Priebatsch attempts to build a game layer on top of the world with his location-based service startup [?]. The adoption of game techniques to non-traditional areas such as finance, sales, and education has become such a phenomenon that the Gartner Group included “gamification” [?] on its 2011 Hype List.

A serious game, defined by Zyda, is “a mental contest, played with a computer in accordance with specific rules that uses entertainment to further government or corporate training, education, health, etc”. The system created by Makahiki consists of a number of smaller games which combined, provides a game playing experience to the participants. So we consider that Makahiki framework produces a serious game, instead of a gamified educational application. While they are different, both gamification and serious games are trying to solve problems by influencing people’s behavior utilizing both intrinsic and extrinsic motivation. Differentiating Makahiki from the other collegiate sustainability competitions are the game design elements in Makahiki.

An Alternative Reality Game (ARG) is one type of serious game related to Makahiki in the way that they both blend real and virtual world activities in the serious gaming context. McGonigal defines ARG as “games you play to get more out of your real life, as opposed to games you play to escape it” [?]. Her award winning serious ARG game “World Without Oil” [?] and “Evoke” [?] are designed with the goal of empowering people to come up with creative solutions to urgent real-world problems. Some players reported that the online game affected their real world behaviors [?].

There are several sustainability related serious games. Similar to games created by Makahiki, Vermontivate [?] is a team-based game where the players compete to accrue as many points as possible for their towns or schools by participating in a variety of sustainability-focused actions. Unlike Makahiki, Vermontivate does not have individual points or prizes and relies more on self-

reported participation.

Reeves et al. described the design of Power House, an energy game that connects home smart meters to an online multiple player game with the goal to improve home energy behavior [?]. In the game, real world energy data are transformed into a “more palatable and relevant form of feedback”, and players may be incentivized by the in-game rewards to complete more energy-friendly real-world behaviors.

RecycleBank’s “Green Challenges” [?] is another serious game that used online gaming techniques to motivate participants to learn about green living and to take small green actions to live more sustainable lives offline. According to the “Gaming For Good” report [?], the challenge showed a 71% increase in unique visitors and 97% of participants surveyed said that the challenge increased their knowledge about how to help the environment.

### 1.3 Game Design and Motivation

The research in serious games and gamification inspires the design of Makahiki in terms of benefits of a game, game design and how the game design can motivate and engage players in serious contexts.

Why games? Results of a study published in the May 1998 issue of Nature [?] demonstrated that video game players experience regular releases of dopamine during game play. Dopamine is a neurotransmitter that signals pleasure rewards for food, sex and addictive drugs, such as cocaine.

Psychology professor Mihaly Czikszentmihalyi introduced a specific kind of happiness that he named “flow” [?], which is considered as one of the fundamental reasons that people play games [?]. Flow is a state of absorption, characterized by intense concentration, loss of self-awareness, a feeling of being perfectly challenged (neither bored nor overwhelmed).

Nicole Lazzaro argued that the use of extrinsic rewards will decrease the motivation to use your products and services once you remove that reward [?]. Vockell resonated that in education psychology, extrinsic motivators may lead to short-range activity increase but reduction in long-range interest in a topic. While intrinsic motivators motivate people best when they are working toward personally meaningful goals [?].

In order to design a game that that is intrinsically motivated, Amy Jo Kim presented “Smart Gamification” which focuses on designing an effective “Player Journey” with intrinsic reward preferred over extrinsic reward [?]. Kim pointed out that intrinsic values are greater than extrinsic rewards and “a good game takes the player on a journey toward mastery”.

The design of Makahiki is influenced by the above game design thinking. It combines extrinsic rewards such as prizes and achievement badges with intrinsic motivation such as learning how to improve our environment. Makahiki employs Kim’s player journey idea [?] to design the on-boarding and progression of levels in the smart grid game. Other game mechanics, such as the “appointment mechanics” and the “social interaction” are also employed in the design of Makahiki.

## 1.4 Serious game Framework assessment

A game framework is “comprised of a collection of different tools, utilities, and interfaces that hide the low-level details of the various tasks that make up a game.”[?] One of the benefits of using a game framework is that, if correctly designed, it will provide useful and reusable “building blocks” with which to develop a variety of serious games. Yet how are we to know if a serious game framework has been “correctly designed”?

In order to assess a serious game framework such as Makahiki, it is important to assess the serious games the framework produces. One fundamental question in evaluating a serious game is the extent to which the game achieves its “serious” purpose. This is quite different from traditional entertainment games, in which evaluation focuses on usability or playability [?]. In the field of serious games, there is an increasing focus on the methodology of game evaluation [?].

De Freitas and Oliver [?] point out that there are few frameworks to support the evaluation of education games. They introduce a four dimensional framework for evaluating educational games and simulations. The framework consists of: the context, the pedagogy, the representation, and the learner (or player).

These approaches focus on evaluation of a single serious game, as opposed to a serious game **framework**. There exists some general purpose assessment tools such as GEQ (Game Engagement Questionnaire)[?] and QUIS (Questionnaire for User Interaction Satisfaction)[?] that are used for

measuring game engagement and user interactions.

During my literature search, I have not found any research in the area of assessment methodology for serious game frameworks. The design of SGSEAM is to provide an assessment method for the particular needs of assessing a serious game framework.

## 1.5 Research Description

The overall research question this thesis investigates is: What forms of information technology infrastructure can support effective and efficient development of serious games for sustainability?

In order to address this research question, I have designed two systems:

- A software infrastructure for development of serious games for sustainability.
- An assessment method that provides evidence regarding the strengths and weaknesses of the software infrastructure for the development of serious games for sustainability.

### 1.5.1 Makahiki

I developed an innovative serious game framework for sustainability called Makahiki, as a software infrastructure for the development of sustainability challenges. Makahiki explores one section of the design space where virtual world game mechanics are employed to affect real world sustainability behaviors. The ultimate goal of the Makahiki project is not just to affect behaviors during the course of the game, but also to produce long lasting, sustained change in behaviors and outlooks by participants.

Makahiki has a unique feature set intended to foster more rapid innovation and development. These features include: (1) an open source license and development model which makes the technology available without charge and facilitates collaborative development and improvement; (2) support for an “ecosystem” of extensible, interrelated, customizable games and activities; (3) real-time game analytics for research and evaluation; (4) pedagogically organized and extensible learning activities; (5) a responsive user interface supporting mobile, tablet, and laptop displays; and (6) support for deployment to the cloud as an inexpensive option for hosting the competition.

The Makahiki framework has been successfully used to create and manage seven real-world sustainability games by four organizations, in different environments and with different requirements. The instances are: 1) the Kukui Cup energy challenges at the University of Hawaii at Manoa in 2011, 2012 and 2014; 2) the HPU Kukui Cup energy challenges at the Hawaii Pacific University in 2012 and 2013; 3) the East West Center energy and water challenge in 2012; and 4) the Kukui Cup educational challenge at the Holy Nativity School in 2013.

### 1.5.2 SGSEAM

While these real world Makahiki instances provide some evidence regarding the ability of the Makahiki framework to support sustainability games in different environments, a more rigorous evaluation of Makahiki would yield better quality insight. There exists some assessment methods for serious games and for general purpose such as system usability. I have not found any research in the area of assessment method for assessing a serious game framework from the aspect of a full lifecycle of a serious game. I designed an assessment method called Serious Game Stakeholder Experience Assessment Method (SGSEAM), to assess a serious game framework from all the stakeholders in a serious game lifecycle.

In a nutshell, SGSEAM (pronounced “sig-seam”) identifies the most important stakeholders of a serious game framework and provides a method for gaining insight into the strengths and shortcomings of the framework with respect to each stakeholders’ needs. We consider SGSEAM as an assessment method instead of an evaluation method. The main purpose of an evaluation is to “determine the quality of a program by formulating a judgement” [?]. An assessment, on the other hand, is nonjudgmental. SGSEAM does not try to judge a framework according to a standard, instead, it is used to identify the major strengths and shortcomings of a framework so that the community can benefit from the assessment by learning from the strengths and improving the shortcomings.

SGSEAM identifies the major stakeholders whose experiences affect a serious game framework as a software infrastructure as: players, system admins, game designers, game managers and game developers. For each stakeholder, multiple *in-vivo* and *in-vitro* assessment approaches can be used,

depending on the resources available. The more approaches applied, the higher one's confidence in the accuracy of the assessment results.

### **1.5.3 Evaluation**

Makahiki, as a serious game framework for sustainability, had been used by 4 different organizations to create a total of 7 serious game instances targeting education and fostering sustainable behavior among the communities. The first Kukui Cup Energy challenge at the University of Hawaii at Manoa (UHM) was held in 2011 for 3 weeks for over 1,000 first year students living in the residence halls. UHM subsequently held the second and third Kukui Cup Energy challenges in 2012 and 2014 for different first year students and different durations, for 9 months and 2 weeks respectively. Hawaii Pacific University (HPU) held their 3-week Kukui Cup Energy challenge in Fall 2012 and 2013 for about 200 students each year. An international organization called the East-West Center (EWC) held a 2-week Kukui Cup Energy and Water challenge for the international residents living in the residence halls. A Hawaii private school called Holy Nativity School (HNS) held a pilot 7-month Kukui Cup challenge for the elementary school students.

The real world case study of Makahiki is to look at the different requirements of these organizations, and how the corresponding different configurations in the Makahiki framework can be done to support these requirements, thus gain insight into how well the Makahiki framework can be used and customized in the real world scenarios.

The successful creation of serious game challenges by four different organizations provides evidence that Makahiki can be successfully tailored to the needs of different organizations. First, UH and HPU used different metering infrastructure, and EWC collected their resource data manually. HNS did not have energy data at all. Second, while UH and HPU challenges involved only energy consumption data, the EWC challenge involved both energy and water consumption data. Third, the IT infrastructure at UH and HPU provided authentication services using CAS (Central Authentication Service) and LDAP, while EWC used UH's CAS and the built-in Django authentication. Fourth, the user interface was customized to “brand” each challenge with the logo, thematic elements, and the education contents of the sponsoring organizations. Lastly, HPU instances were

hosted on its local infrastructure, while UHM, EWC and HNS were hosted on the Heroku PaaS cloud. More details of the real world instance evaluation results are described in [Section 6.1](#) of the Results chapter.

In addition to the real world instance evaluation, I carried out a formal assessment of the Makahiki framework by applying the SGSEAM to Makahiki to determine the strengths and weaknesses of the framework from the five serious game stakeholders' experience perspectives.

Following the SGSEAM process, the assessment plan ([Appendix B](#)) was created as the first deliverable of the process, which guided the SGSEAM assessment of Makahiki. The assessment plan identifies the participants in the categories of five SGSEAM stakeholders. Two categories of assessment approaches were determined. The *in-vivo* assessment approaches of pre-post effectiveness study, in-game surveys and post-hoc interviews were applied to the real-world Kukui Cup challenges created by the Makahiki framework, to assess the experiences of players, system admins, game designers, and game managers. The *in-vitro* assessment approaches of in-lab study used the UHM ICS691 serious game course in Spring 2013 to assess the experiences in game installation, game design and game development.

After creating the assessment plan, I carried out the Makahiki assessment according to the SGSEAM plan, a data repository ([Appendix C](#)) was created to collect all the assessment data. The data was analyzed to determine the strengths and weaknesses of the Makahiki framework. The results of the analysis are described in the [Section 6.2](#) of the Results chapter. The final deliverable of the SGSEAM assessment process was created as the improvement action report described in [Appendix D](#).

The SGSEAM assessment report for Makahiki ([Appendix D](#)) lists the strengths and weaknesses of Makahiki, and the recommended improvement actions from the five stakeholders' perspectives. SGSEAM revealed that the major strengths of Makahiki are the ability to create an engaging and effective serious game for an organization and the ability to provide an easy-to-use interface for game designers and managers. SGSEAM revealed the major weaknesses of Makahiki are difficulty in developing new enhancements to extend the framework from a game developer's perspective, difficulty to integrate external services such as LDAP and email server from a system admin's per-

spective, and the lack of a WYSIWYG content authoring tool from a game designer's perspective.

The application of SGSEAM to Makahiki has effectively identified several strengths and weaknesses of the Makahiki as the serious game framework for sustainability.

## 1.6 Contribution

The contributions of this research are:

### 1.6.1 Makahiki: A serious game framework for sustainability

The Makahiki framework creates a unique type of serious game that combines both data on resource consumption and game content related to sustainability and represents an innovation in the area of serious game and sustainability education. Makahiki enables different organizations to easily create customized sustainability education and behavior change serious games that can be deployed in both local and cloud infrastructures. Makahiki is an open source system hosted on GitHub.

### 1.6.2 SGSEAM: a serious game framework assessment method

The SGSEAM contributes to the emerging field of serious game framework evaluation research. It describes a method for assessing serious game frameworks from the stakeholder experience perspective. The goal of SGSEAM is to identify the major strengths and weaknesses of a serious game framework, and provide the improvement actions to the framework developers. The benefits of SGSEAM assessment are that the developers of serious game frameworks can learn and improve from the findings of the assessment.

### 1.6.3 Insights into the strengths and weaknesses of Makahiki

The application of SGSEAM to Makahiki identified several strengths and weaknesses of Makahiki as a serious game framework for sustainability. The strength and weakness report and improvement action report deliverable of this research provides valuable inputs to Makahiki developers for improving the framework, thus benefiting the community and organizations that use Makahiki as a tool for sustainability education and behavior change.

#### **1.6.4 Insights into creating and running a variety of real-world serious games**

Makahiki contributed to the creation and running of seven real-world sustainability games by four organizations. These game instances engaged over 3,770 students from 4 educational institutes. These games were created in various configurations and hosting environments. The major differences involve the duration of the challenge, the population that could participate in the challenge, the type of resource(s) such as energy or water, whether they have smart meters installed, and the type of server hosting. Additional differences in requirements include the type of the authentication for participation and differences in game mechanics. The successful creation of these serious game instances provides evidence that Makahiki can be successfully tailored to the needs of different organizations.

#### **1.6.5 Insights into managing cloud based serious games**

Hosting and managing the 4 Makahiki instances from 3 organizations in the cloud provided interesting experiences with cloud-based serious games. The successful use of Makahiki in the Heroku cloud platform provided a good alternative for organizations such as EWC and HNS who have less technology capacity. It also provides greater flexibility in infrastructure scaling up and down depending on the usage demand. In the context of serious games, which are often running for a period of time and have various usage demand throughout the game period, using the cloud infrastructure is a good alternative.

#### **1.6.6 Insights into the strengths and weaknesses of SGSEAM**

The application of SGSEAM to Makahiki effectively identified the strengths and weaknesses of Makahiki and provided improvement suggestions to Makahiki developers. It also revealed the strengths and weaknesses of SGSEAM as an assessment method. The major strengths of SGSEAM includes the identification of the five categories of serious game stakeholders, a clearly defined three-step process and corresponding deliverables, more complete coverage of serious game life cycle than other methods, and the flexibility of choosing from a set of additive assessment approaches. The major weaknesses of SGSEAM includes the process being expensive and time consuming, the

small sample size of the post-hoc interview approaches. Despite the limitations, the application of SGSEAM can still result in useful insights into the strengths and weaknesses of a serious game framework.

## 1.7 Outline

The dissertation is organized into the following chapters:

- Chapter 2 looks at related research, including serious game, gamification, serious game framework, and framework assessment.
- Chapter 3 describes the design and implementation of the Makahiki system.
- Chapter 4 describes the serious game framework assessment method SGSEAM.
- Chapter 5 describes evaluation of both Makahiki and SGSEAM.
- Chapter 6 describes results of the evaluations.
- Chapter 7 concludes the dissertation with a list of contributions and future directions.
- Appendix A contains the publications that have come out of this research that I have authored or co-authored.
- Appendix B describes the SGSEAM assessment plan for Makahiki.
- Appendix C describes the SGSEAM data repository for the Makahiki assessment.
- Appendix D describes the SGSEAM improvement action report as the result of the Makahiki assessment.
- Appendix E describes the SGSEAM assessment guide written specifically for Lucid's serious game framework.
- Appendix F contains the survey questionnaire that was administered in the game to assess the player's experiences.

- Appendix G describes the google forms that were used in the in-lab experiment studies.

## **CHAPTER 2**

### **RELATED WORK**

This chapter examines research related to sustainability, serious games and gamification, game design, serious game frameworks, and framework assessment. [Section 2.1](#) and [Section 2.2](#) discusses the needs for sustainability education and the current implementation of collegiate sustainability competitions. [Section 2.3](#) discusses related work on serious games and recent development in gamification. [Section 2.4](#) looks at the applications of “serious game” in the sustainability context. Finally, [Section 2.6](#) and [Section 2.7](#) examine the serious game framework and its assessment.

#### **2.1 Sustainability and Education**

Sustainability is defined as “a requirement of our generation to manage the resource base such that the average quality of life that we ensure ourselves can potentially be shared by all future generations”[?]. The depletion of natural resources has become a major focus of governments, organizations and citizens all over the world. There is widespread support for a dramatic increase in energy efficiency and change in energy use from fossil fuels to renewable sources[?].

In Hawaii, the need for transition is especially acute, as the state leads the US both in the price of energy and reliance on fossil fuels as an energy source. Almost 90% of the energy used in Hawaii is from imported fossil fuel, primarily oil, [?], as shown in [Figure 2.1](#). Launched by the US Department of Energy and the State of Hawaii, the Hawaii Clean Energy Initiative [?] aims to achieve 70% clean energy by 2030 with 30% from efficiency measures, and 40% coming from locally generated renewable sources.

Government action is required to change the balance between methods of generation and consumption. On the other hand, in terms of personal change for a sustainable lifestyle, “green” actions addressing the responsibility of consumers such as recycling, and saving energy and water need to be pursued [?].

Changing people’s behavior with respect to energy holds significant promise in reducing energy use. Darby’s survey of energy consumption research found that identical homes could differ in

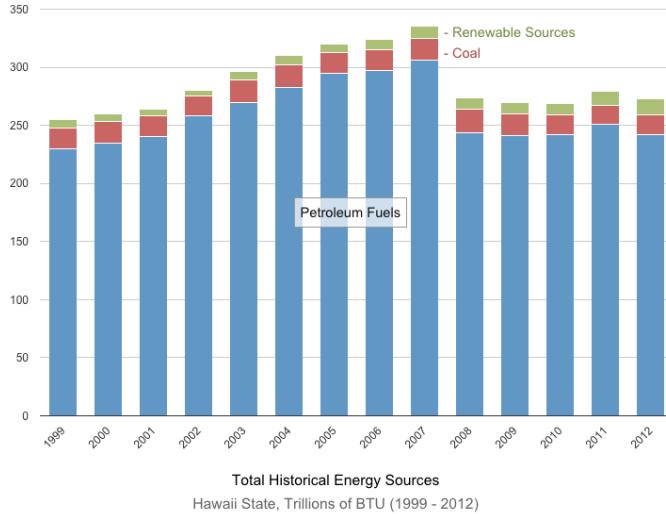


Figure 2.1: Hawaii Energy Sources [?]

energy use by a factor of two or more [?]. Data from a military housing community on Oahu show energy usage for similar homes can differ by a factor of 4 [?].

## 2.2 Collegiate Sustainability Competitions

The creation of Makahiki is largely motivated by the needs of colleges and universities to run competitions for sustainability education. Energy and water competitions or challenges have been introduced to college dormitories and residential homes as ways to facilitate and incentivize resource reduction to achieve sustainability goals. A survey by Hodge [?] found that there are 163 college residence hall energy competitions taking place or being planned for the 2010–2011 academic year in North America to engage students in sustainability issues. Hodge found that the average reduction in electricity use during these competitions is 9%. In 2014, 109 schools participated in the Campus Conservation Nationals (CCN) [?], a nationwide electricity and water use reduction competition for colleges and universities. Overall, the competition resulted in 4.5% average electricity competition reduction at the building level, with the top 10 schools achieving reductions ranging from 11% to 24%.

A basic type of sustainability competition uses a “minimal tech” solution such as a web page

and manual posting of data and results on a periodic basis such as once a week. The Harvard University Green Cup Competition [?] and the Wellesley College Green Cup [?] are examples of this type of competition. During the 2012 Harvard University Green Cup Competition, there were a 4.3% average electricity reduction across the participating residence houses and an individual house reduction of 16.6%, as shown in [Figure 2.2](#).

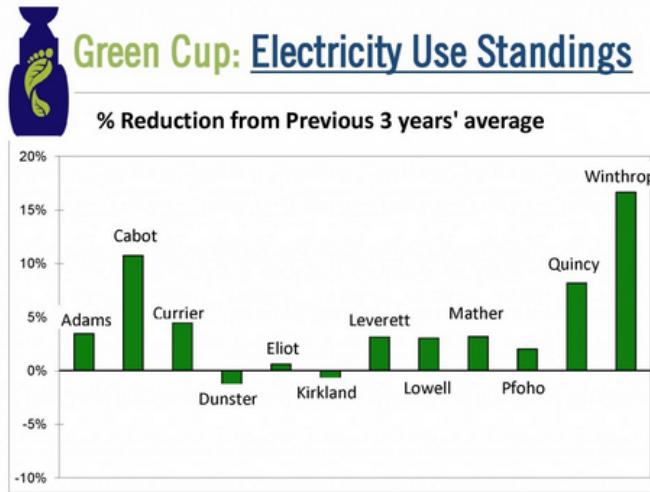


Figure 2.2: Harvard Green Cup Competition for Energy

Some organizations built their own custom in-house solutions to support the infrastructure need for sustainability competitions. Oberlin College dorm energy competition [?], Duke University's Eco-Olympics competition [?], and Western Washington University's Go for Green Challenge [?] are examples of this type of competition. Petersen et al. [?] describe their experiences deploying a real-time feedback system in an Oberlin College dorm energy competition in 2005 that included 22 dormitories over a 2-week period. The competition used an automated data monitoring system that was developed in-house to provide dormitory residents with real-time web-based feedback on energy and water use. They found a 32% reduction in electricity use across all dormitories and claimed that real-time resource feedback system combined with education and incentives can motivate and empower college students to reduce resource use in dormitories. [Figure 2.3](#) shows the design of the Oberlin system.

Universities such as University of British Columbia [?] and Bowdoin College[?] chose to out-

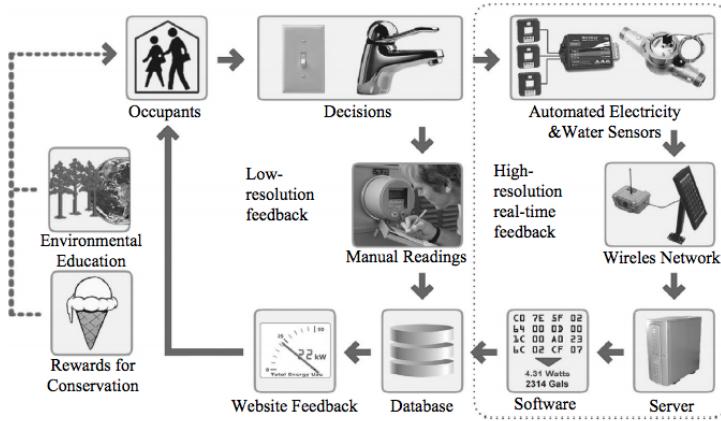


Figure 2.3: Oberlin College’s Custom Designed Sustainability Competition [?]

source the technology to a commercial provider for their sustainability competition. University of British Columbia started the energy competition titled “Do it in the Dark” [?] in November 2010 for the 6 first year student residence houses. During the competition, the participating residence reduced overall energy consumption by 16.3%. The competition was hosted by Lucid Design Group’s Building Dashboard platform [?], which provided online real-time feedback of energy usage for the participating residence, and other social interactions such as sharing on Facebook through the web interface. [Figure 2.4](#) shows the interface of the University of British Columbia Energy competition using Lucid’s Building Dashboard platform.

Campus Conservation Nationals (CCN) [?] is a nationwide electricity and water use reduction competition for colleges and universities. Now in its fifth year, hundreds of universities participated in the competition across North America. In 2014, 109 schools participated in the Energy Competition, which includes 1,330 buildings on the school campuses and 265,000 students and staff actively involved in the 3 week competition. Overall, the competition resulted in a 4.5% average electricity competition reduction at the building level, with the top 10 schools achieving reductions ranging from 11% to 24%. The software infrastructure used in CCN is supported by Lucid Design Group through its the Building Dashboard platform, which is used by the participating schools to get feedback from their energy and water use, compare performance and track the competition standings.

In summary, these collegiate sustainability competitions all have the goal of engaging students in



Figure 2.4: University of British Columbia Energy Competition[?]

sustainability issues using resource feedback and incentives such as prizes. Some competitions also provide educational content or activities such as sustainability related events. Some competitions include a simple point scheme enabling participants to win points for their teams or residence dorms by participating in events or reducing consumption. The team or dorm with the most points wins a prize. Some competitions also include a social sharing feature such as sharing their commitments to sustainability on Facebook as a way to enforce and encourage the positive behaviors. **Table 2.1** compares some of the prominent competitions with respect to these features.

School	Feedback	Education/ Activities	Points	Social	Infrastructure
Harvard University	manual				simple webpage
Wellesley College	manual				simple webpage
Duke University	manual	✓	✓		custom website
Western Washington University	manual	✓	✓		custom website
Oberlin College	real-time	✓		✓	custom website
Bowdoin College	real-time	✓		✓	out-source
University of British Columbia	real-time	✓		✓	out-source
Campus Conservation Nationals	real-time	✓		✓	out-source

Table 2.1: University energy competitions

None of these choices are ideal: the custom in-house solution requires sophisticated design and

implementation skills; out-sourcing can be financially expensive and impedes evolution; and the simple webpage solution does not fully leverage the possibilities of advanced information technology.

The goal of Makahiki is to provide an open source framework for different organizations to create engaging sustainability serious games, including this kind of sustainability competitions. In addition to improvements to all of features discussed above, Makahiki creates a game-like interface such as tracking individual points and game plays, level progression and raffle game, and other features to be discussed later in [Chapter 3](#). Makahiki lowers the overhead to those who would build a custom in-house solution by providing pre-built components. It can lower the financial cost to those who would out-source by providing an open source alternative. Finally, it provides an opportunity for those who would choose a minimal tech solution to instead provide more sophisticated information technology.

## 2.3 Serious Games and Gamification

The game design elements in Makahiki differentiate Makahiki from other collegiate sustainability competitions. We consider the sustainability competitions created by the Makahiki framework as a type of serious game. A serious game is “a game designed for a primary purpose other than pure entertainment” [?]. It includes categories such as educational games and advergames (advertising), political games, and training games (also known as game-learning). Zyda [?] defines a serious game as “a mental contest, played with a computer in accordance with specific rules that uses entertainment to further government or corporate training, education, health, etc”.

One prominent example is Foldit [?], a multiplayer online game that helps solve problems that computers cannot solve very well. In this case, online gamers around the world were able to do what biochemists have been trying to do for a decade: decipher the structure of a protein that is key to the way HIV multiplies. [Figure 2.5](#) shows a screen shot of the Foldit game.

An Alternative Reality Game (ARG) is one type of serious game related to Makahiki in the way that they both blend real and virtual world activities in the serious gaming context. McGonigal defines ARG as “games you play to get more out of your real life, as opposed to games you play

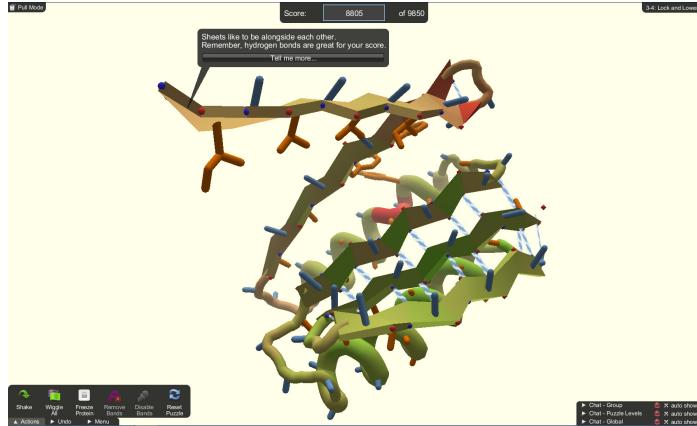


Figure 2.5: Serious Game Example: Foldit is solving a serious problem [?]

to escape it” [?]. Her award winning serious ARG game “World Without Oil” [?] and “Evoke” [?] are designed with the goal of empowering people to come up with creative solutions to urgent real-world problems. [Figure 2.6](#) shows an screen shot of the World Without Oil game. The game started on April 2007 for 32 days and concluded on June 2007. Approximately 2000 players participated in the game. They were asked to imagine what life would be like without oil, or try actually living without oil, then post their stories and thoughts in the forms of letter, emails, voicemails, blog posts, online videos and images. Some players reported that the online game affected their real world behaviors in some ways[?].

ARGs have also been used to support learning. Connolly et al. [?] discuss the development of an educational ARG to motivate secondary school students across Europe to learn foreign languages. The results of the pilot run of the game in 2009 indicated that 92% of students felt the game motivated them to learn a second language. One of problems the team identified is the limitation of the Moodle [?] platform the game is based on and there is potential to improve the effectiveness of the game.

The report of the ARGOSI project [?] provides insights into the use of ARGs in game based learning and the challenges they face in the field of higher education. The pilot was run at the University of Bolton with the aim of providing an engaging alternative to traditional methods of introducing students to university life. Adoption of the game was fairly low with 173 players of which only 23 (13%) were active. The project identifies a number of questions surrounding educational



Figure 2.6: Serious Game Example: World Without Oil - Play it before you live it. [?]

ARGs, such as motivation, relationship to curriculum, marketing and timing. The report suggests that a complete ARG model may not be appropriate for wholesale learning, but there is certainly potential in using game elements.

Makahiki shares similarities with ARGs in that they both combine real and virtual world activities to achieve “serious” purposes, be it educational or behavior changing.

While “Serious Games” has been an active research topic for decades, “Gamification”, on the other hand, is a relatively new subject. Deterding et al. [?] defines gamification as “the use of game design elements in non-game contexts”. The term only came into widespread use starting in 2010 [?]. Gartner [?] predicts that by 2015, more than half of companies managing innovation processes will employ gamification, applying game mechanics to application areas including productivity, finance, health, sustainability, news, user-generated content and e-learning.

Deterding et al. [?] describes the distinctions between gamification, serious games and other related concepts, as shown in [Figure 2.7](#). According to Deterding, a) Gamification is about games. It is different than playful interaction or playful design. b) Gamification uses game elements. It is not a complete game such as a serious game. c) Gamification applies to non-game contexts.

Similar to serious games, gamification uses games for other purposes than its normal expected use for entertainment. d) Gamification focuses on design. It is not game-based technology or practice of wider game ecology.



Figure 2.7: Serious Game and Gamification (source: Deterding [?])

The system created by Makahiki consists of a number of smaller games which combined, provides a game playing experience to the participants. So we consider that the Makahiki framework produces a serious game, instead of a gamified educational application. While they are different, both gamification and serious games are trying to solve problems with game thinking. Gamification's main driving force is motivation, similarly, serious games also try to solve the motivation problem and influence people's behavior for "serious" purpose. Bosch [?] considered the serious game Foldit as a successful example of gamification in science.

FourSquare [?] is probably the most recognized example of applying game mechanics to a location-based networking application. It is a location-based, game-like service where players check-in to locations for virtual points and rewards. By employing gamification elements such as points, badges, levels and leader boards, it engages users to revisit a location such as a restaurant or pub and become a loyal customer and finally the "mayor" of the place. Certain virtual rewards such as the "mayor" of a particular Starbucks can be converted into real products, e.g. a free coffee. Foursquare proved that simple game mechanics can affect user behavior by engaging 10 million

customers with a successful business model. [Figure 2.8](#) shows the home page for Foursquare.

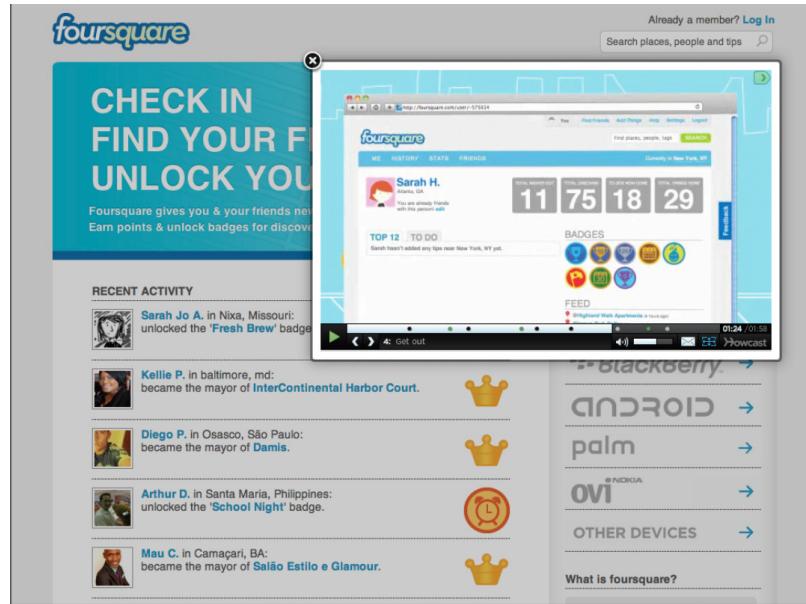


Figure 2.8: Gamification Example: Foursquare makes modern badges popular[?]

Nike+ [?] is a social running game that employs game mechanics to encourage runners - both casual and hardcore - to compete and improve their fitness, with the goal of solving the main problem of most fitness programs: motivation. Nike+ allows runners to upload their exercise data to its web site, and start challenging themselves and their friends. They can also get support from their friends through the web site. The game makes running and exercise fun, which eventually serve the “serious” purpose of making people healthy. [Figure 2.8](#) shows the home page for Nike+.

RibbonHero [?] is a game that helps users discover new Microsoft Office features, as shown in [Figure 2.10](#). The goal is to have users build familiarity and expose them to the Office UI, so that they understand what kinds of features are available. According to the creator of the game, Office “has a lot of powerful features that users might not know but can be really useful”. The game gives users a chance to learn those features through a game interface, rather than reading the software manuals or watching the typically dry IT training videos.

A persuasive game is another genre of video game that uses game design to influence player behaviors. In his book “Persuasive Games, The Expressive Power of Video games”, Ian Bogost[?] argues that persuasive games have a unique power that goes beyond other forms of computational

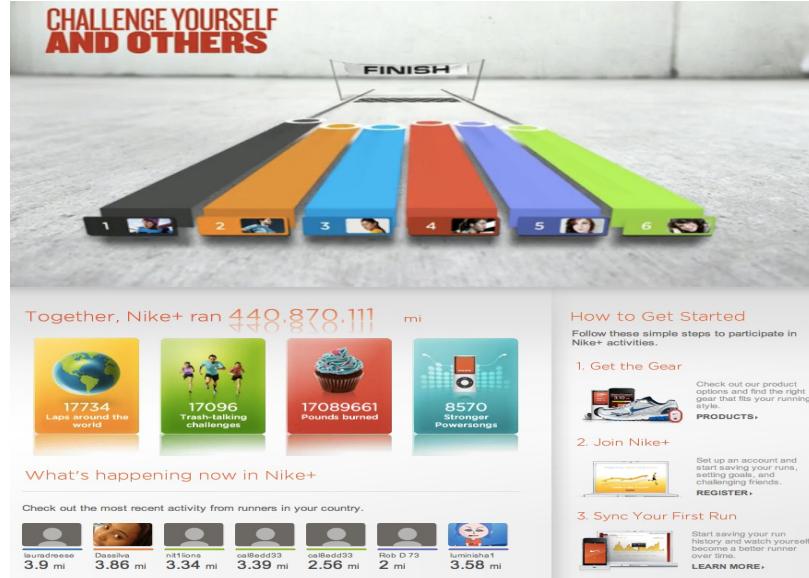


Figure 2.9: Gamification Example: Nike+ makes fitness run[?]

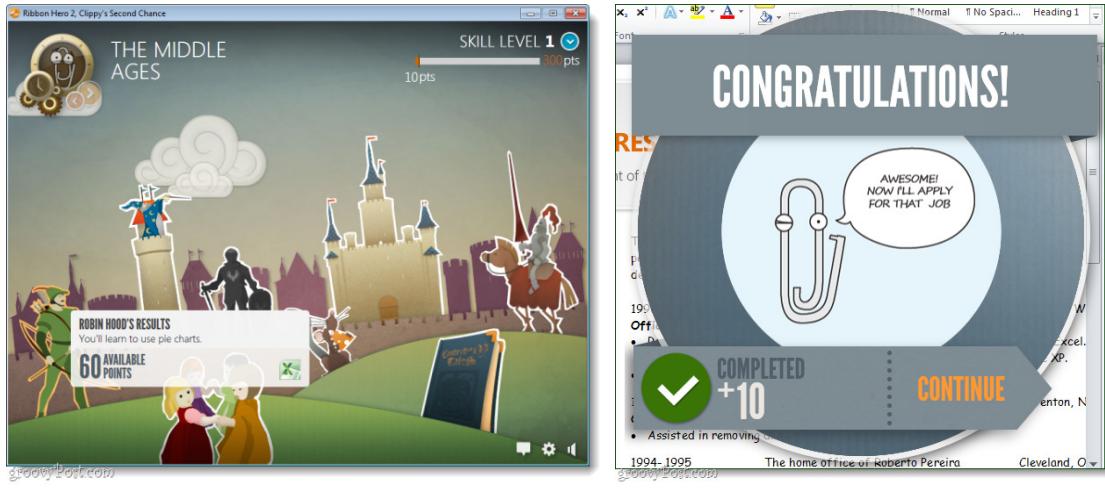


Figure 2.10: Gamification Example: RibbonHero Helps to Learn Office[?]

persuasion. Not only can persuasive games support existing social and cultural positions, as in serious games, but they can also disrupt and change those positions, leading to potentially significant long-term social change. Persuasive games are closely tied to “Persuasive Technology”[?], designed to change attitudes or behaviors of the users through persuasion and social influence, but not through coercion. Loren Baxter[?] posted that persuasive design, the use of psychology in design

to influence behavior, could benefit UX design.

Interactive design can also be applied to achieve sustainability goals. One example is the “SmartGauge” dashboard in Ford’s hybrid cars[?] (Figure 2.11(a)), where a digital plant responds to how energy-efficient the users driving behavior is. The design gives drivers a game, with the goal to grow more lush and beautiful leaves, along with a visual reward, by driving efficiently and thus promotes a more environmental behavior. Similarly, the design of “Piano Staircase” [?] (Figure 2.11(b)), created by Volkswagen Sweden, installed in a metro station in Stockholm, is to make the staircase next to the escalator look and respond like a piano keyboard, so that every step on the stair will generate different piano sounds every time a commuter walks on it. Observation indicates that 66 percent more people chose to play the “piano staircase” game over using the escalator. It is a good example of gameful design for persuading and encouraging energy-efficient behavior.

## 2.4 Serious Games for Sustainability

Makahiki is a framework for sustainability games. Serious games for sustainability are games designed to achieve the goal of sustainability development.

Vermontivate [?], a type of community sustainability game, shown in Figure 2.12, is similar to the games created by Makahiki. Vermontivate is a team-based game where the players compete to accrue as many points as possible for their towns or schools by participating in a variety of sustainability-focused actions. It runs for six weeks with participants mainly drawn from the state of Vermont. Because of the differences between the team size, team scores are calculated “per-capita” by adding up a team’s total points and dividing by its number of players. A new set of challenges is announced every week with a different theme such as team-building, food, energy, transportation, capital, and future action. According to the initial sponsor of the game, Vermont Energy Investment Corporation, the first Vermontivate game began in May of 2012 and attracted 225 players from 31 Vermont towns. 95% of players reported average to above-average understanding and engagement with climate change and sustainability after playing Vermontivate, compared to 78% prior to playing [?]. Unlike Makahiki, Vermontivate game does not have individual points or prizes, instead its focus is on activities and relies on self-reported participation.



(a) Efficiency Leaves



(b) Piano Stair vs. Escalator

Figure 2.11: Gameful Design for Sustainability

Figure 2.12: Vermontivate Sustainability Game (source: Vermontivate [?])

Similar to Vermontivate, RecycleBank's "Green Challenges" [?] is another serious game that used online gaming techniques to motivate participants to learn about green living and to take small

green actions to live more sustainable lives offline. Figure 2.13(a) shows a web page from the “Green your home Challenge” game. According to the “Gaming For Good” report [?], 49,000 individuals participated in the “Green Your Home Challenges”. Figure 2.13(b) shows a section of the survey results regarding the self-reported sustainability behavior changes after the game. Partnered with Google Analytics and ROI research, they found that:

- Gamification can increase awareness of positive environmental actions. 97% of participants surveyed said the game increased their knowledge of the environment.
- Games can drive individuals to take positive social and environmental actions. Most participants surveyed indicated they are very or extremely likely to take green actions as a result of participating in the challenge.
- Games are an effective and appealing educational tool. 86% participants agreed online games and contests can be a good way to inform and educate them personally.



(a) Green Your Home Challenge

What green actions do you take?	Pre	Post	% +
I turn off the lights	18%	26%	44%
I use CFL/Eco bulbs	28%	38%	36%
I conserve water/energy	34%	45%	32%
I buy local produce	0%	14%	----
I wash clothes in cold water	0%	7%	----

(b) Game Change Behavior

Figure 2.13: RecycleBank - Gaming for Good

The Opower Social Energy Application [?] is another kind of sustainability serious game, available as a Facebook app on both web and smartphones. It is developed in partnership with Facebook and the Natural Resources Defense Council (NRDC) with the intention of making saving energy social [?]. [Figure 2.14](#) shows a screenshot of the game. Through the app, participants can compare their energy use to similar homes, share energy saving tips, compete with friends, and participate in team-related energy reduction challenges. The app lowers the adoption barrier by directly importing energy usage data from Opower utility accounts. If the participant does not have an Opower utility account, he can enter data manually from his utility bills, which requires more effort and motivation to participate. Compared to Makahiki, the energy usage feedback from Opower is not real-time (monthly), and the home energy saving tips or recommendations are not linked to points, badges, or other virtual or real rewards.

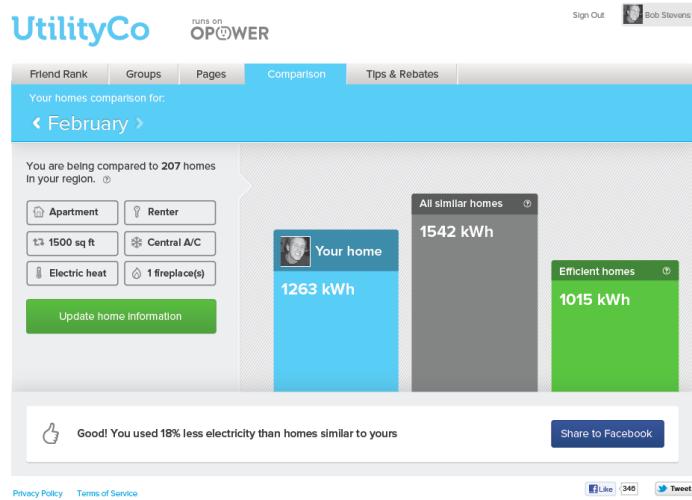


Figure 2.14: Social Energy Application Engaging Consumers (source: Opower [?])

Reeves et al. [?] described the design of Power House, an energy game that connects home smart meters to an online multiple player game with the goal of improving home energy behavior. In the game, real world energy data is transformed into a “more palatable and relevant form of feedback,” and players may be incentivized by the in-game rewards to complete more energy-friendly real-world behaviors. [Figure 2.15](#) shows a screenshot of the PowerHouse game. The games created by Makahiki share a similarity with PowerHouse in that they both provide real-time energy feedback to the player. Makahiki has more educational activities and less simulation game play than

PowerHouse.



Figure 2.15: Power House Game to Save Energy(source: Reeves [?])

Another simulation based sustainability game is EnerCities [?] developed by Dutch game developer Paladin Studios with a 1.4M budget funded by the European Commission. Awarded the title of “Best Learning Game 2010”, it offered an online learning game for young people (typical target group: 15-20 years old) to experience energy-related implications. The goal of the game is to create and expand virtual cities dealing with pollution, energy shortages, renewable energy, etc. Available both as a standalone website and on Facebook, The game was played by thousands of students from more than 110 schools across Europe. The game offers a semi-realistic simulation with cartoony 3D (via Unity3D plug-in[?]) visual styles with multiple level game play. [Figure 2.16](#) shows a screenshot of the game. Similar to PowerHouse game, EnerCities game engages player via the graphical simulation of the energy related issues. The player experience is more video game-like than the other sustainability games currently available in Makahiki. On the other hand, since Makahiki is an extensible and configurable game framework, it is possible to include simulation games such as EnerCities and PowerHouse inside Makahiki to provide better player experience and engagement.

[Table 2.2](#) summaries the above serious games for sustainability and compares them to the games created by Makahiki.



Figure 2.16: EnerCities - simulation based sustainability game(source: EnerCities [?])

Game	Type	Targeted player	Rewards	Competition	Mobile
Vermontivate	Website	Community	Prizes	Team	✓
RecycleBank	Website	Community	Prizes	Team	
Opower	Facebook App	Energy consumer	Virtual	Individual	✓
PowerHouse	2D Simulation	Energy consumer	Virtual	Individual	
EnerCities	3D Simulation	Schools	Virtual	Individual	
Makahiki	Website	Schools	Prizes & virtual	Team & individual	✓

Table 2.2: Serious games for sustainability

## 2.5 Game Design and Motivation

The research in serious games and gamification inspires the design of Makahiki in terms of benefits of a game, game design and how the game design can motivate and engage players in serious contexts.

Why games? Results of a study published in the May 1998 issue of Nature [?] demonstrated that video game players experience regular releases of dopamine during game play. Dopamine is a neurotransmitter that signals pleasure rewards for food, sex and addictive drugs, such as cocaine.

A favorite subject of Greek vase paintings in the ancient games exhibition in the British Museum's department of Greek and Roman antiquities is Ajax and Achilles playing a kind of board game called backgammon, as illustrated in [Figure 2.17](#). It is noteworthy that both Ajax and Achilles have full armor on while playing the game. According to Arthur A. Krentz [?], in Plato's "Republic", the term "paideia" (in Greek, means education/culture), "paidia" (means play/game/pastime/sport), and "paides" (means children), have the same root. The three terms often show up in the same context. Krentz states that "The central aim of pedagogy (paidagogia) is to encourage learning as a form of play (paidia), which is the most persuasive and effective approach to learning" .

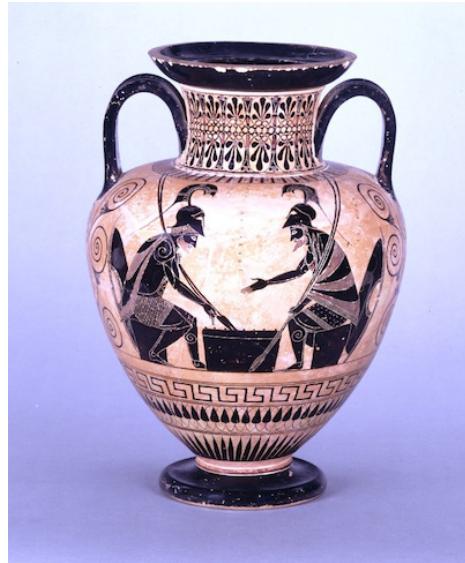


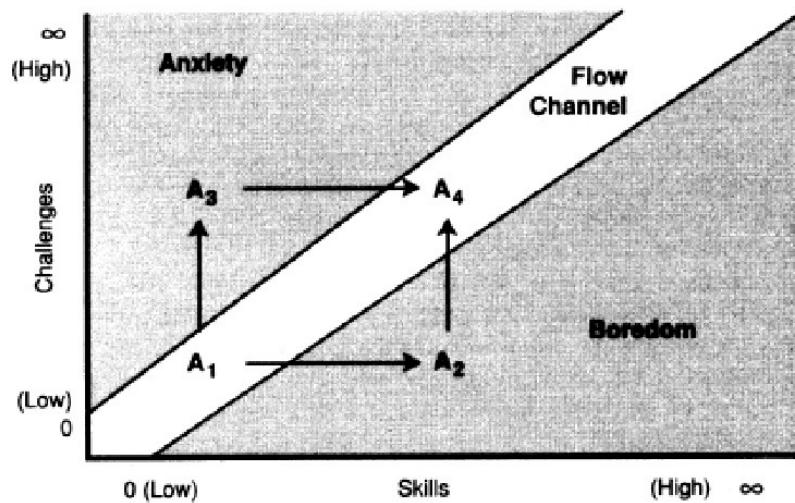
Figure 2.17: Ancient Games Shown in British Museum

Moving forward a couple of millennia, World of Warcraft (WoW) is a massively multiplayer online role-playing game (MMORPG) with 11.1 million subscribers, currently the world's most popular MMORPG. Nick Yee [?] pointed out that the collaborative nature of most activities makes MMORPG unique. "It's the people that are addictive, not the game". "Most importantly, it is the reward of being socialized into a community of gamers and acquiring a reputation within it" . He claimed [?] that "WoW truly is a virtual Skinner box", smoothly increasing reward and difficulty and reinforcing player commitment along the way.

In her popular and inspiring TED talk "Gaming can make a better world" [?] and in her book "Reality is Broken" [?], researcher and game designer Jane McGonigal makes a case for why good

games make us better, and how they can help us change the world. She notes that more than 3 billion hours a week is spent playing video games by our society. She says that the average gamer plays 10,000 hours of games by age 21. That's about the same number of hours that students spend in high school and middle school. There are 500 million gamers today, playing on all sorts of platforms from the iPhone to game consoles. Instead of the common conception that gaming is a waste of time, she argues that "playing games is the single most productive thing we can do with our time" and that it is the solution to our "Broken Reality". Byron Reeves also argues in his book "Total Engagement" [?], that games, especially MMO type games, can change the way people work and businesses compete.

Psychology professor Mihaly Czikszentmihalyi introduced a specific kind of happiness that he named "flow" [?], which is considered as one of the fundamental reasons that people play games [?]. Flow is a state of absorption, characterized by intense concentration, loss of self-awareness, a feeling of being perfectly challenged (neither bored nor overwhelmed) and a sense that time is flying. In order to achieve flow, the important condition is a balanced goal that is challenging yet achievable within the individual's ability. This balance is referred to as the flow channel as shown in [Figure 2.18](#).



[Figure 2.18: The state of flow is achieved between anxiety and boredom \[?\]](#)

In order to understand why people play games, Richard Bartle identified four player personality types by studying players of Multi-User Dungeon (MUD) games in 1960s [?]:

1. **Achievers**: driven by in-game goals, usually some form of points gathering - whether experience points, levels, or money.
2. **Explorers**: driven to find out as much as they can about the virtual construct - including mapping its geography and understanding the game mechanics.
3. **Socializers**: use the virtual construct to converse and role-play with their fellow gamers.
4. **Killers**: use the virtual construct to cause distress on other players, and gain satisfaction from inflicting anxiety and pain on others.

Bartle's player type model has been the basis for understanding the player motivation. Dan Dixon presented the limitation and misuse of Bartle's model in general games and gamification contexts [?]. Amy Jo Kim applied the model in her gamification approach by overlaying social actions from the game on top of the player types [?], as shown in [Figure 2.19](#).

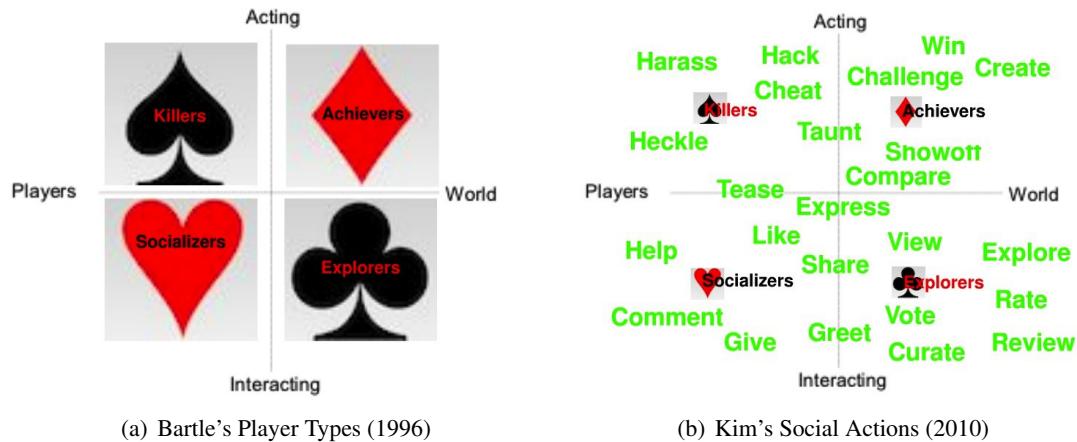


Figure 2.19: Player Types

Different game mechanics and elements can be used to serve different functions in satisfying players' needs, and the basic elements such as points, badges, and leader boards are the defining attributes of the current gamification practices [?]. [Figure 2.20](#) illustrates these basic game mechanics and elements.

Seth Priebatsch [?] stated that you can get anyone to do anything with 7 game dynamics. Techcrunch [?] published a “secret” game dynamics play deck that is used by Priebatsch’s com-

	Reward	Status	Achievement	Self Expression	Competition	Altruism
Points	●		●		●	●
Levels		●	●		●	
Challenges	●	●	●	●	●	●
Virtual Goods	●	●	●	●	●	
Leaderboards		●	●		●	●
Gifting & Charity		●	●		●	●

(a) Satisfies Human Needs (source: Bunchball[?])



(b) Basic Mechanics (source: Deterding [?])

Figure 2.20: Basic game mechanics and elements

pany SCVNGR. The play deck is a set of 47 flash cards. Each card illustrates one game dynamics. SCVNGR employees are instructed to memorize them and apply them in their applications as needed.

There are many debates and criticism over whether gamification itself is inherently good or bad. Designer Stephen Anderson claimed that gamification mistakes extrinsic rewards (rather than intrinsic motivation) for the power of games and hence offers only feedback, not goals & rules [?].

Jane McGonigal spoke about her concern about current state of gamification in the GDC 2011 talk titled “We don’t need no stinking badges: How to reinvent reality without gamification” [?]. She argued that current gamification confuses intrinsic/extrinsic motivation and proposed “Gameful Design” instead of “Gamification”. She claimed that “Gameful is player-oriented”, which presumed

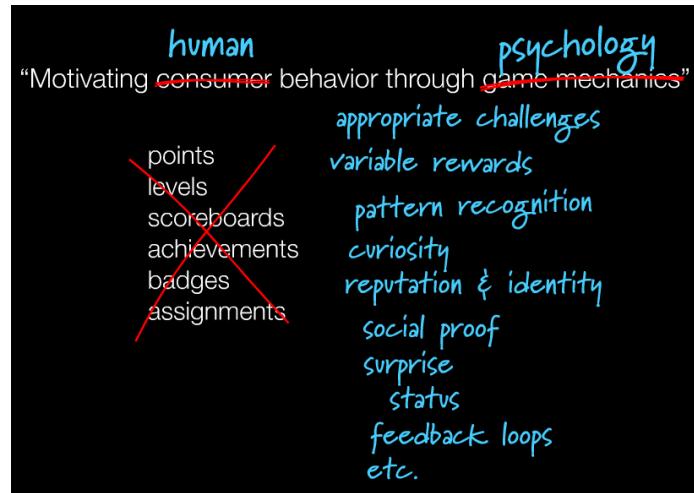


Figure 2.21: Gamification is about extrinsic rewards (source: Anderson [?])

that the loyalty program type gamification is product or service oriented. While the current gamification is about extrinsic reward, with points, badges, and levels, gameful design is about intrinsic reward, with positive emotion, relationships, meaning and accomplishment.

Nicole Lazzaro argued that the use of extrinsic rewards will decrease the motivation to use your products and services once you remove that reward [?]. Vockell reasoned that in educational psychology, extrinsic motivators may lead to a short-range activity increase but a reduction in long-range interest in a topic. In contrast, intrinsic motivators motivate people best when they are working toward personally meaningful goals [?].

Michael Wu argues that extrinsic rewards can jumpstart intrinsic motivation [?]. He claimed that gamification just has to work long enough for some other processes to take over as the primary driver of value. Subsequently, it becomes a secondary reinforcement system.

In order to design a game that is intrinsically motivated, Amy Jo Kim presented “Smart Gamification” which focuses on designing an effective “Player Journey” with intrinsic reward preferred over extrinsic reward [?]. Kim pointed out that intrinsic values are greater than extrinsic rewards and “a good game takes the player on a journey toward mastery.” As illustrated in Figure 2.22(a), when over time players progress from newcomer to regular and finally to enthusiast, they progress from novice to expert to master. Kim also incorporates the MDA framework [?] by using it to guide and motivate the player journey as illustrated in Figure 2.22(b).

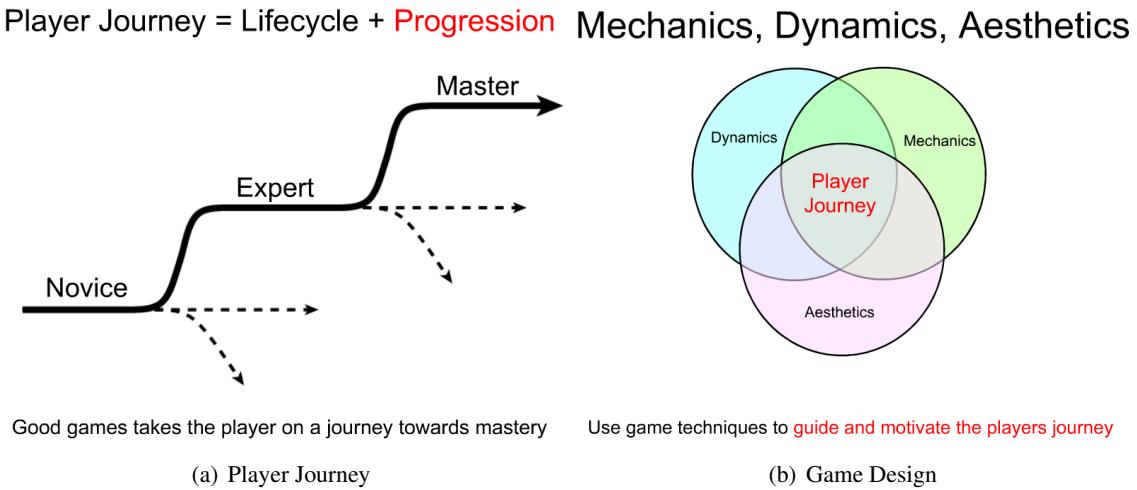


Figure 2.22: Designing Player Journey (source: Kim [?])

Similarly, Sebastian Deterding not only criticized the current practice of simple gamification practices but stressed the importance of “meaningful play” and proposed a user experience design around the three most important aspects: Meaning, Master and Autonomy [?]. It is an adaptation of the three elements that motivate people in Daniel Pink’s book “Drive: The Surprising Truth About What Motivates Us” [?]. Deterding explained that the reason why we play is because of the meaning and autonomy (choices) in the game. The mastery in the game give us fun and enjoyment.

The design of Makahiki is influenced by the above game design thinking. It combines extrinsic rewards such as prizes and achievement badges with intrinsic motivation such as improving our environments by learning. Makahiki employs Kim's player journey idea [?] in the design of on-boarding and progression of levels in the smart grid game. **Table 2.3** lists some of the most commonly used game mechanics compiled by Gamification.org[?] and how they are used in Makahiki.

## 2.6 Serious Game Frameworks

Game frameworks (also known as game engines) [?] are “comprised of a collection of different tools, utilities, and interfaces that hide the low-level details of the various tasks that make up a game”. Examples of game frameworks include:

- Unreal [?]: The Unreal Engine is a game engine developed by Epic Games, it is primarily

Mechanics	Description	Player Type	In Makahiki
Levels	a system of reward for a cumulation of points, unlocked as players progress	Achievers	level
Points	a running numerical value given for any single action or combination of actions.	Achievers	point
Achievements	badge, completed something	Achievers	badge
Progression	success is granularly displayed (progress bar) through the process of completing tasks	Achievers	
Appointment Dynamics	at a predetermined times/places a user must return for a positive effect	Archivers	approval process
Countdown	players are only given a certain amount of time to do something	Achievers	
Quests	a journey of obstacles a player must overcome	Achievers, Explorers	
Reward Schedules	the fixed or variable time frame and delivery of the rewards	Achievers	daily goal game
Loss Aversion	the player having to perform an action to avoid losing	Achievers	event sign up penalty
Lottery	the winner is determined by chance	Achievers, Explorers	raffle game
Status	the rank or level of a player	Achievers, Socializers, Killers	
Community Collaboration	rally community to work together to solve a riddle, a problem or a challenge	Socializers	social bonus, Facebook share

Table 2.3: Game mechanics and use in Makahiki

used in first person shooter games, providing tools and building blocks for 3D rendering, collision detection, AI, networking etc.

- PapayaMobile [?]: PapayaMobile is a free cross platform social game engine on Android and iOS platform. It provides an SDK and a platform for mobile game developers to create and release games in a “user-friendly, straightforward way”.
- OpenLabyrinth [?]: OpenLabyrinth is an open source game framework that allows its users to create, run and analyze a wide range of different pathway-based activities for healthcare education.
- Fabula [?]: Fabula is an open source Python game engine for adventure, role-playing and

strategy games and digital interactive storytelling. It provides a library and game world abstraction intuitive to people who have not been involved in game development before and hide as much as low level technical details as much as possible.

One of the benefits of using a game framework is that, if correctly designed, it will provide useful and reusable “building blocks” with which to develop a variety of games. Similarly, serious game frameworks also provide building blocks that enable the serious game developer to focus more time and thought on content and results instead of on the technical details and infrastructure for creating the serious game.

There are two serious game frameworks related to sustainability development. One such framework is the Building Dashboard [?], developed by Lucid Design Group, as shown in [Figure 2.23](#).



Figure 2.23: Building Dashboard (source: Lucid [?])

Building Dashboard is commercial platform that “enables energy reduction competition and empowers building occupants to become active participants in energy management”. It is used to support the Campus Conservation Nationals (CCN) [?], a nationwide electricity and water use reduction competition on college campuses. In CCN 2014, the framework was used by 109 schools in North America to display the energy and water consumption of the competition participants. It enables viewing, comparing and sharing building energy and water use information on the web through a visual interface.

Building Dashboard is similar to Makahiki that they are both frameworks for supporting sustainability competitions, but the cost of Building Dashboard as a commercial system creates a barrier to wider adoption. In addition, the Building Dashboard solution focuses on providing energy information as a passive media. Besides a scoreboard, there is little interaction between participants and the system. There are no game elements other than providing a scoreboard to display the ranking of the competing teams, moreover, there is no individual points or ranking. Unlike Makahiki, Building Dashboard does not have the concept of individual registered player account and personalization, thus it does not have the capability to provide the evidence of individual player engagement.

**Table 2.4** compares the features of Building Dashboard and Makahiki.

	<b>Building Dashboard</b>	<b>Makahiki</b>
Real-time feedback	✓	✓
Competition	✓	✓
Social	✓	✓
Commitments	✓	✓
Scoreboard	✓	✓
Points		✓
Integrated Educational Content		✓
Pluggable games		✓
Personalization		✓
Game analytics		✓
Open Source		✓

Table 2.4: BuildingDashboard and Makahiki framework comparison

A game framework is used to create serious games. **Figure 2.24** illustrates a full lifecycle of a serious game using a framework. By instantiating from a serious game framework, the system admins create a new serious game with the default configurations. The next phrase is for the game designers to configure the instance according to their organization's specific needs. In the case of the built-in features not able to meet the organization's need, the game can be extended by the game developers to add new features or customizations. After the game instance is finalized, the system admins will deploy it to the infrastructure of the choice of the organization. Once the game instance is deployed and running, the next phrase is for the game managers to manage the game using the admin interface provided by the framework, and the players will play the game. The last phrase of the lifecycle is the analysis phrase after the game is over where game managers, designers and

developers can evaluate and learn from the results of the game.

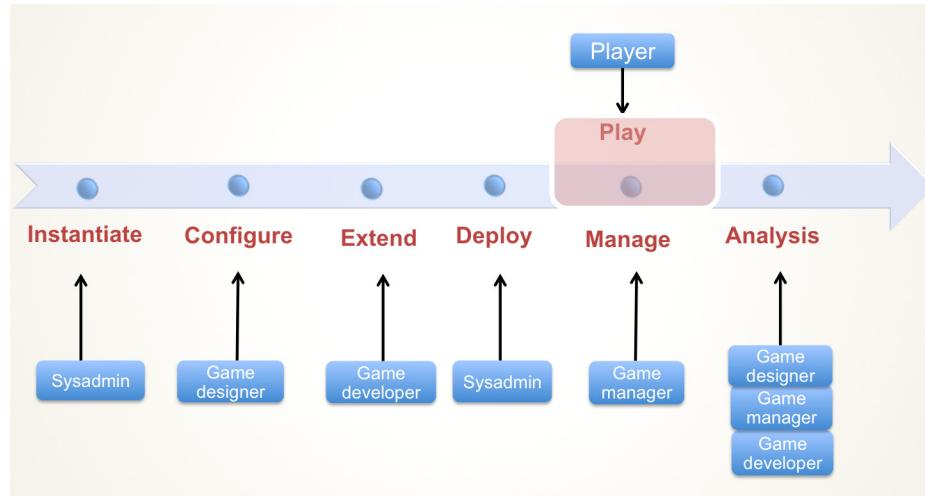


Figure 2.24: Serious game lifecycle using a framework

The design of Makahiki follows closely the serious game lifecycle using a framework, which provides building blocks and utilities for each phrase of the lifecycle. Lucid's Building Dashboard system only provides a pre-built Software As a Service (SaaS) solution where the game designers can choose from a small sets of dashboard widgets and game managers can enter energy data manually if needed. There is no option for system admin to install the system locally or deploy to the infrastructure of their choices, or for the developers to extend the game. There is no game analytics provided in BuildingDashboard. Makahiki has a much richer feature set than Lucid's Building Dashboard.

Another framework related to sustainability is the Stanford Energy Services Platform [?], as shown in Figure 2.25. It provides services to support the creation of energy efficiency program and research. The services include data storage, a recommendation system, user registration and participation assignment, surveys and analytics. It has been utilized to support the implementation of several of Stanford's energy saving projects and energy related serious games, such as the Power House game, Power Down game, and Energy Calculator. At this point, there is not enough information about the Stanford Energy Services Platform regarding the availability and features.



Figure 2.25: Stanford Energy Services Platform (source: Stanford [?])

## 2.7 Serious Game and Framework Assessment

This section examines the assessment of serious game frameworks. It starts by looking at the assessment of serious games, then the assessment of game frameworks in general. From my literature search, I have not yet found any prior work concerning a comprehensive approach for the particular needs of a serious game framework assessment. Nevertheless the research in game and framework assessment methods provides ground works for the SGSEAM method that we designed for use in assessing a serious game framework.

In order to assess a serious game framework such as Makahiki, it is important to assess the serious games the framework produces. One fundamental question in evaluating a serious game is the extent to which the game achieves its “serious” purpose. This is quite different from traditional entertainment games, in which evaluation focuses on usability or playability [?]. In the field of serious games, there is an increasing focus on the methodology of game evaluation [?].

De Freitas and Oliver [?] point out that there are few frameworks to support the evaluation of educational games. They introduce a four dimensional framework for evaluating educational games and simulations. The framework consists of: the context, the pedagogy, the representation, and the learner (or player). Figure 2.26(b) illustrates the evaluation framework. Harteveld [?]

also agrees that “Evaluatory research for games with a serious purpose is still at its infancy”. He proposes an evaluation framework called “Triadic Game Evaluation (TGE)” for assessing serious games. It consisting of three perspectives: Reality, Meaning, and Play, as illustrated in the Figure 2.26(a). Both frameworks are used to evaluate the effectiveness of serious games, not serious game frameworks.

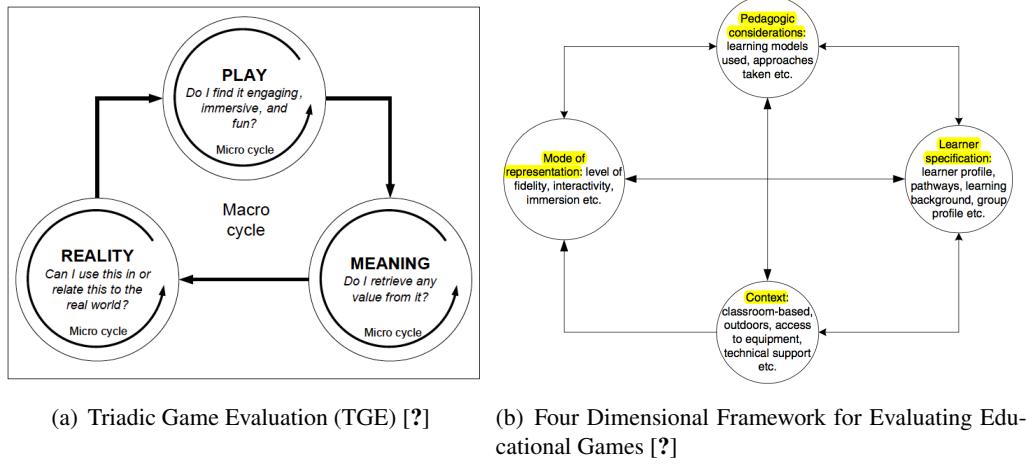


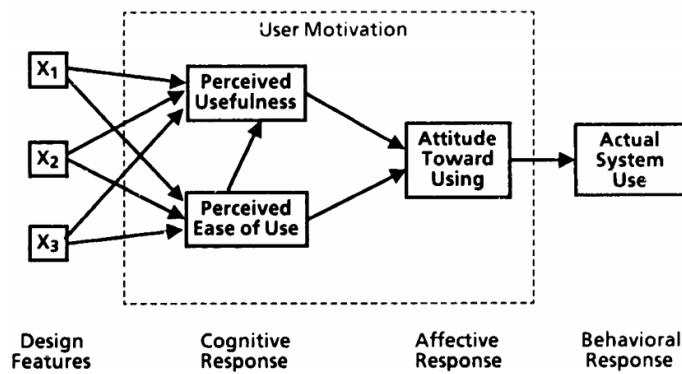
Figure 2.26: Evaluation Framework for Serious games

Effectiveness assessment is often part of the serious game evaluation. In their evaluation of the EnerCities game, Knol and Vries reported that they conducted a survey to test differences in awareness concerning energy-related issues between a group who had actually played the game and those who had not (between-participants design). Examples of the survey questionnaires include:

- What did you find out about energy saving and “green energy” after playing the game?
- After playing the EnerCities game I was interested in learning more about energy saving and “green” energy
- Playing the Enercities game has increased my concern about the environment
- Playing the Enercities game made me aware of the linkages between economy, energy usage and environment
- Playing the EnerCities game made me aware that I should lower my own energy usage

The above approaches focus on evaluation of a single game, as opposed to a game *framework*. One of the benefits of using a game framework is that, if correctly designed, it will provide useful and reusable “building blocks” with which to develop a variety of games. Yet how are we to know if a game framework has been “correctly designed”?

Berger and Muller [?] describe their approach of using the Technology Acceptance Model (TAM) to evaluate the custom game engine Fabula [?]. Technology Acceptance Model [?] is a well received theoretical model on assessing user acceptance of computer-based information systems, introduced by Fred Davis in his doctoral thesis in 1985. TAM considers that system use is a response that can be predicted by user motivation, which is influenced by an external stimulus of the system’s features and capabilities. [Figure 2.27](#) illustrates the original Davis model. X1, X2 and X3 in the figure represent the system features.



[Figure 2.27: Technology Acceptance Model \(TAM\) \[?\]](#)

As Chittur [?] points out in his review of TAM, there is skepticism among some researchers regarding the rigor of the model. There exists other assessment tools such as Game Engagement Questionnaire (GEQ) and Questionnaire for User Interaction Satisfaction (QUIS).

Questionnaire for User Interaction Satisfaction (QUIS) [?] is a usability assessment tool developed in the HCI lab at the University Of Maryland, College Park. It is designed to assess user’s subjective satisfaction regarding the human/computer interface of software systems. Currently licensing is required to access the QUIS questionnaires.

Another usability assessment tool is the usability metrics described in Tullis and Albert ’s book

“Measuring the User Experience” [?]. For a usability procedure about completing transactions, Tullis and Albert suggest measuring task success, user efficiency, issues-based metrics, self-reported metrics, and live website metrics. Task success is a simple metric for a given task, does the user complete it or not? User efficiency is a measurement of the effort required for the user to complete the task. For example, we can measure this by the amount of time spent to complete the task. Issues-based metrics involve measuring the number of times usability issues are encountered. Self-reported metrics are based on user responses to survey questions. Finally, live website metrics can be derived from analyzing the logs created by the website to understand the user experience.

The above assessment methods focus on one or two aspects of the lifecycle of a serious game using a framework. [Figure 2.28](#) illustrates the existing assessment methods in the context of a full serious game lifecycle using a serious game framework.

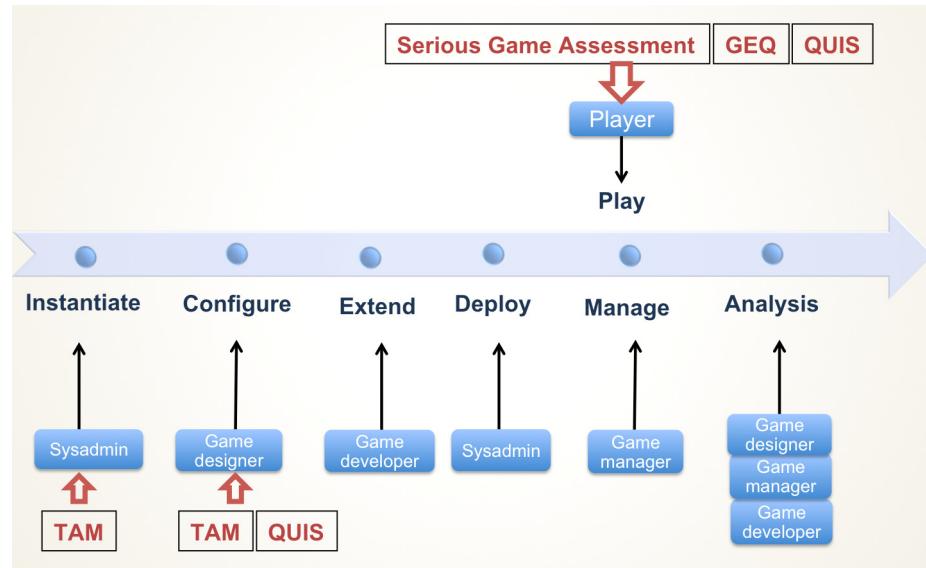
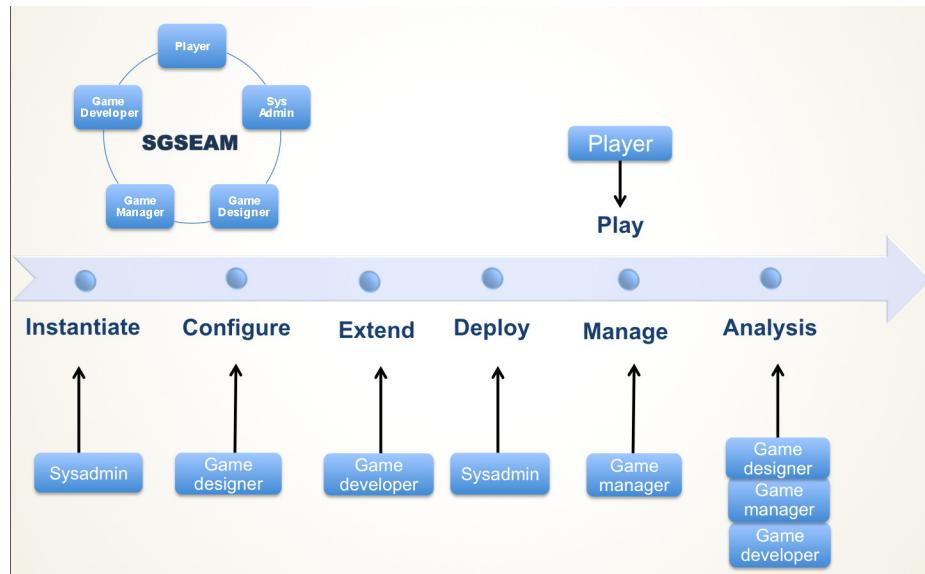


Figure 2.28: Existing assessment methods in a serious game lifecycle

The serious game assessment methods only assess the player aspect of the lifecycle, where the player effectiveness is the main concern of the individual serious game assessments. The Game Engagement Questionnaire (GEQ) only assesses the engagement from the player’s perspective. The Technology Acceptance Model (TAM) focuses on the perceived acceptance of the framework in the case of system admins and game designers’ perspective. The Questionnaire for User Interaction

Satisfaction (QUIS) only assesses the usability of players and game designers when using the game and the framework.

The design of SGSEAM, the Serious Game Stakeholder Experience Assessment Method, is to address all the stakeholders in a serious game lifecycle using a serious game framework. As illustrated in [Figure 2.29](#), SGSEAM assess all the stakeholders in the full lifecycle of a serious game to understand the strengths and weaknesses of a serious game framework from these stakeholders' perspectives.



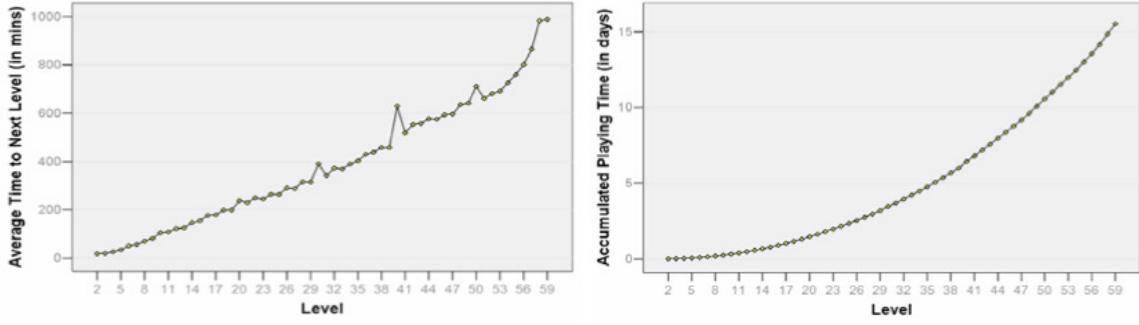
[Figure 2.29: Existing assessment methods in a serious game lifecycle](#)

### 2.7.1 Game Metrics

Game metrics can be as important as creativity in game design. As Nadia Oxford points out, in the game industry, player metrics collection and analysis are widely practiced to allow game designers to determine what the player audience likes and dislikes about a certain game experience [?].

Ducheneaut et al. provides an example of using game metrics to analyze player's experience in a quantitative approach [?]. They reported the relationship of playing time and leveling in the MMORGs, as shown in [Figure 2.30](#):

Matt Fairchild [?] and Kontagent [?] lists and explains some of the social games metrics as



(a) Average time required to reach a level

(b) Average accumulated play time by level

Figure 2.30: Player Metrics (source: Ducheneaut [?])

shown in [Table 2.5](#).

Metrics	Description
DAU	Daily Active Users, is the number of active users over the course of a single day.
MAU	Monthly Active Users, is the total number of users in a given month.
DAU/MAU ratio	Comparing Daily Active Users to Monthly Active Users shows roughly how many days per month the average user engages with a game.
ARPU	Average Revenue Per User, is measured as total revenue divided by the number of users. ARPU can be broken down by type of revenue, day, country, demographic, etc.
Churn	Turnover rate (or “attrition rate”) of active players.
K Factor	= (Infection Rate) * (Conversion Rate). An Infection Rate is how much a given user exposes the game to other players, such as through status updates or email invites. A conversion rate is when that “infection” results in a new sign up. K factor measures the viral effect of a game. A high K Factor indicates effectiveness of bringing in new players.
Engagement	Measures how long users spend playing a game. How many features do they access? How many pages does the average user view? What percentage are returning visitors?

Table 2.5: Social Game Metrics

[Appdata.com](#) gathers independent application metrics from most of the social game applications. For example, the graphs in [Figure 2.31](#) shows the DAU (Daily Active User) and MAU (Monthly Active User) metrics for the popular FarmVille [?] social game [?]:

While the QUIS [?] is used for measuring user interaction satisfaction in a general software

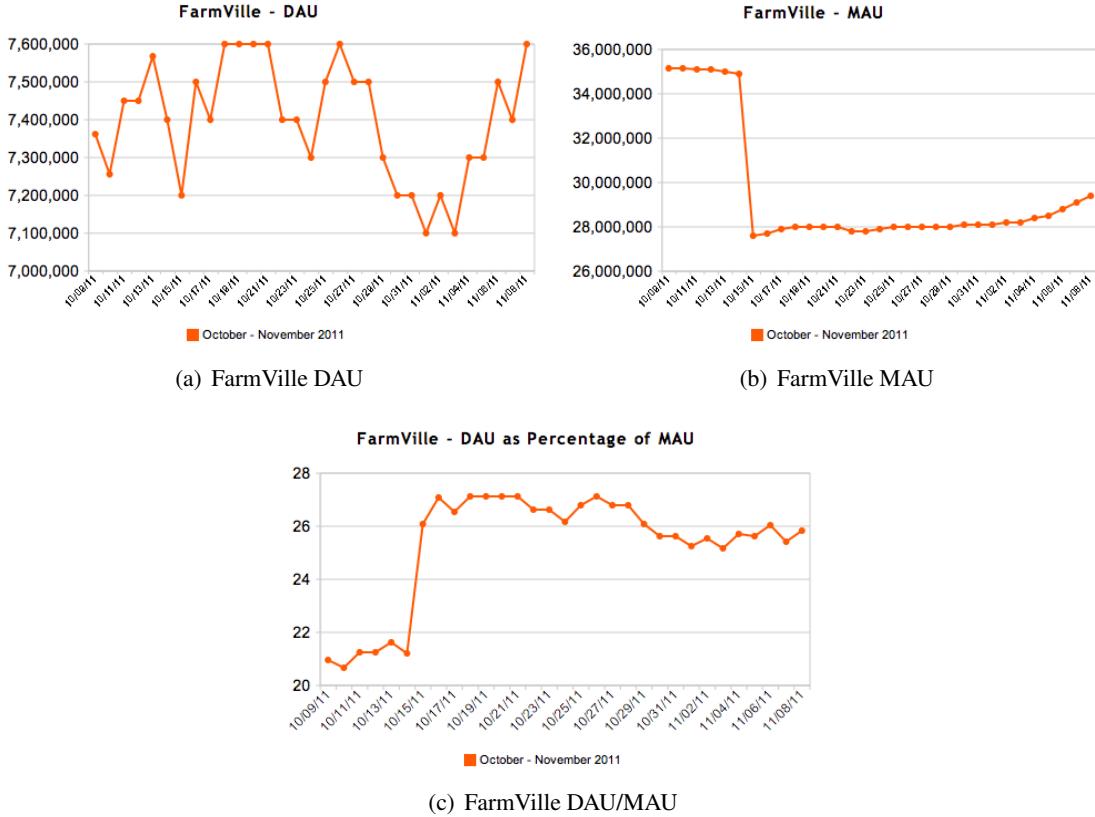


Figure 2.31: Social Game Metrics Example(source: Appdata.com [?])

system, Game Engagement Questionnaire (GEQ) [?] developed by Brockmyer et al. is used to assess the engagement in a game. The questionnaire provides a “psychometrically” strong measure of levels of engagement specifically while playing video games. While the GEQ could measure the engagement level of positive game experience, the original intent of the research is to “examine risk and protective factors for negative game impact”. Figure 2.32 shows the questionnaire items.

## 2.8 Summary

In summary, this chapter discusses related work on serious games and gamification, game design thinking, serious games for sustainability, serious game frameworks and their assessment. The motivation of Makahiki is to provide a low cost infrastructure that enables the creation and managing of collegiate sustainability competitions. The design of Makahiki is inspired by current research in

1	I lose track of time
2	Things seem to happen automatically
3	I feel different
4	I feel scared
5	The game feels real
6	If someone talks to me, I don't hear them
7	I get wound up
8	Time seems to kind of stand still or stop
9	I feel spaced out
10	I don't answer when someone talks to me
11	I can't tell that I'm getting tired
12	Playing seems automatic
13	My thoughts go fast
14	I lose track of where I am
15	I play without thinking about how to play
16	Playing makes me feel calm
17	I play longer than I meant to
18	I really get into the game
19	I feel like I just can't stop playing

Figure 2.32: Game Engagement Questionnaire (GEQ) items [?]

game design in the context of both serious games and gamification. Serious game evaluation is an emerging field. There exists several assessment methods for serious games and for general purpose such as system usability. During my literature search, I have not found any research in the area of assessment method for assessing a serious game framework from the aspect of a full lifecycle of a serious game. The design of SGSEAM is to provide an assessment method for the particular needs of assessing a serious game framework from all the stakeholders in a serious game lifecycle.

## CHAPTER 3

# MAKAHIKI DESIGN

This chapter describes the design of the Makahiki, a serious game framework for sustainability. It starts with an overview of Makahiki in Section 3.1, followed by a description of the Makahiki system in Section 3.2. Section 3.3 describes the architecture of Makahiki, followed by the design features that make Makahiki an innovative serious game framework.

### 3.1 Overview

Makahiki is an open source “serious game framework for sustainability.” It provides a framework for creating serious games for the purpose of education and behavioral change regarding energy, water, food, and waste generation and use.

The initial version of Makahiki (Version 1) was originally developed as a custom built software solution [?] for the Kukui Cup Energy Competition [?] at the University of Hawaii. The first version of Makahiki provided the following functionalities to support energy challenges at the University of Hawaii: a) A synergistic mixture of real-world and virtual world activities to raise consciousness and literacy regarding energy issues; b) Real-time feedback on energy consumption by residence hall teams; c) Incentives in the form of prizes and raffle games. d) Social networks, both physical (residence hall teams) and virtual (Facebook).

The current version of Makahiki (Version 2) builds upon the prior version with the purpose of providing a serious game **Framework** for sustainability that enables different organizations to easily create customized serious games in the context of sustainability education and behavioral change. In addition to the features from version 1, the current Makahiki framework includes the following features:

- The ability to tailor system functionality to support the requirements of different organizations.
- The ability to support sustainable resource challenges such as water, food, and waste in addition to energy.

- The ability to extend the framework with new mini-games and modules to support different requirements.
- The use of HTML5/CSS3 “responsive” design techniques for support of laptop, tablet, and smart phone interfaces.
- Real-time game analytics to help assess the impact of game mechanics during challenges.
- Support for PaaS (Platform as a Service) facilities such as Heroku. This enables organizations to create and deploy challenges without obtaining physical hardware and its requisite IT support.

The design of Makahiki follows closely the serious game lifecycle using a framework as illustrated in [Figure 2.24](#). In Makahiki, the system admins install or create a new default game instance by instantiating from the Makahiki framework; then the game designers configure or customize the instance according to their organization’s specific needs. If needed, the game developers extend the instance by adding new game widgets. The system admins deploy the finalized instance to the infrastructure of the choice of the organization, either locally or to the cloud. Once the game instance is deployed and running, the game managers can manage the game and the players play the game. In Makahiki, the game analytics are available realtime during the game and after the game.

The next section describes Makahiki as it is viewed by the user roles in a serious game lifecycle using the Makahiki framework.

## 3.2 System Description

This section describes Makahiki from the perspectives of different user roles. It first describes the installation process from the system administrator’s view, followed by the descriptions of different system interfaces to players, game designers, game managers and developers.

### 3.2.1 System Admin Interface

This section describes the system admin interface of the Makahiki system from the perspective of system administrators. It includes the software installation which is part of the game instantiation, and the deployment of the instance.

The design goal Makahiki installation and deployment is to make it an easy process to support different IT infrastructure. Different organizations can choose the best installation and deployment methods that are suitable for them depending on their infrastructure and IT support capacity.

Makahiki supports two forms of installation and deployment: local (on your own machine) and cloud-based (to the Heroku application hosting service). Organizations can install Makahiki locally if they wish to host the system themselves. This requires sufficient hardware resources and IT support to do the installation, perform backups, and monitor the system during the challenge and deal with any outages that occur.

Organizations can instead choose to host Makahiki with Heroku[?], a popular cloud-based Platform as Service (PAAS). Like most cloud services, this type of installation incurs a cost depending on the service level, but has the benefit that no hardware or IT resources are required.

**Table 3.1** outlines the installation steps for both local and cloud environments.

Installation Step	Local	Cloud
1. download software	git clone or download from website	git clone
2. Install tools and dependencies	Python, C compiler, Pip, Virtual environment, Python Imaging Library, Memcache, PostgreSQL	Heroku client
3. Initialize a server instance	initialize_instance	initialize_instance –heroku
4. (Optional) Configure SSL	install webserver, enable SSL	addons:add ssl
5. Start up the server	manage.py run_gunicorn	automatically started in previous step

Table 3.1: Installation process

The main difference between local and cloud installation is step 2, “Install tools and dependencies”. For a local install, all the tools and dependencies will need to be installed manually by system

administrators, including the C compiler, Pip and Virtual environment, Python imaging library, memcache as the caching system, and finally the PostgreSQL database, as well as the configuration of the database. This requires adequate skills from system administrators. On the other hand, cloud installation only requires the installation of a single Heroku client. In the case of cloud installation, all Makahiki dependencies including the PostgreSQL database are satisfied by the instance created in the Heroku cloud, thus minimizing the involvement of system administrators.

Some organizations may want to use HTTPS / SSL to secure communication between the player browser and the website. In the local installation scenario, the system administrator can install a web server such as Apache or Nginx, then configure and enable HTTPS with SSL, and finally configure to route web requests from the web server to the Django application process. Although a fairly standard web server configuration, this local installation step requires a certain level of skill and effort. In contrast, running a Heroku client command “heroku install addons:add ssl” enables the SSL add-on for the application when using cloud-based hosting. For more details, please see the Makahiki installation guide [?] chapter 2.

After the instance is created, the system admin can use the system admin interface to configure the system related settings, such as authentication, wattdepot server connectivity, and email server.

[Figure 3.1](#) shows the system admin interface.

### **3.2.2 Game Designer Admin Interface**

Game designers use the “Challenge Design” admin interface to design and configure the game related settings. There are two categories of settings, one is the challenge settings, the other is the game settings.

The challenge settings define the global configuration of the challenge such as the name and logo of the challenge, the round definition and the point system. The player settings are also defined in this section, as well as other configurable settings such as page, help, templates and the smart grid game content library.

The game settings define the configurations for each individual games and game mechanics in Makahiki. [Figure 3.2](#) shows the challenge design interface.

**Change global setting**

Home > Challenge\_mgr > Global settings > Kukui Cup/UH

**Authentication**

Choose the type or types of authentication for players

**Use cas auth**

Use CAS authentication ? (Optional)

The URL for CAS authentication service. Example: https://login.its.hawaii.edu/cas/ (Optional)

**Cas auth text:**

A button will appear if there is other auth method(s) selected. This is the text for the CAS auth button. Uses [Markdown](#) formatting.

**Use ldap auth**

Use LDAP authentication ? (Optional)

The URL for LDAP authentication service. Example: ldap://localhost:10389 (Optional)

**Ldap search base:**

The search base for the ldap service. Example: ou=users,ou=system (Optional)

**Ldap auth text:**

A button will appear if there is other auth method(s) selected. This is the text for the LDAP auth button. Uses [Markdown](#) formatting.

**Use internal auth**

Use internal authentication ? (Optional)

A button will appear if there is other auth method(s) selected. This is the text for the internal auth button. Uses [Markdown](#) formatting.

**WattDepot server for real time energy data**

If using WattDepot for energy data, provide the WattDepot Server's URL.

**Wattdepot server url:**

The URL for Wattdepot service. Example: http://localhost:8194/wattdepot (Optional)

**Email**

If you want email notifications enable email and provide the host information.

**Email enabled**

Enable email ? (Optional)

**Contact email:**

The contact email of the admin.

**Email host:**  **Email port:**

The host name of the email server. (Optional) The port of the email server

**Email use tls**

Use TLS in the email server ? (Optional)

Figure 3.1: System Admin Settings Page

This page is where the Challenge Designer sets up the Challenge.

1. Edit the Challenge Settings. Next determine the number and length of the Rounds. Then edit the Score Settings.
2. Create the Groups, there must be at least one Group. Then the Teams, and finally the individual players.
3. Check out the PageInfo items to determine which pages are displayed in the Challenge. Update any of the Help Topics and Notice Templates.
4. (Optionally) Edit the Scheduler information. The default settings should be good.
5. Edit the Game settings, enabling or disabling the games and setting their specific settings.

Challenge Settings		Game Settings	
Click on the links below to configure the challenge.		Click on the links below to configure game related settings or administrate the games.	
<a href="#">Challenge</a>	<a href="#">Energy Game</a>	<a href="#">Global settings</a>	<a href="#">Energy goal settings</a>
<a href="#">Round settings</a>	<a href="#">Water Game</a>	<a href="#">Point rubrics</a>	<a href="#">Water goal settings</a>
<a href="#">Players</a>	<a href="#">Smart Grid Game</a>	<a href="#">Groups</a>	<a href="#">Bonus points</a>
<a href="#">Groups</a>	<a href="#">Actions</a>	<a href="#">Teams</a>	<a href="#">Column names</a>
<a href="#">Teams</a>	<a href="#">Levels</a>	<a href="#">Users</a>	<a href="#">Top Score Game</a>
<a href="#">Users</a>	<a href="#">Prizes</a>	<a href="#">Other Settings</a>	<a href="#">Raffle Game</a>
<a href="#">Other Settings</a>	<a href="#">Raffle prizes</a>	<a href="#">Page infos</a>	<a href="#">Participation Game</a>
<a href="#">Page infos</a>	<a href="#">Participation settings</a>	<a href="#">Help topics</a>	<a href="#">Quest Game Mechanics</a>
<a href="#">Help topics</a>	<a href="#">Quests</a>	<a href="#">Notice templates</a>	<a href="#">Badges</a>
<a href="#">Notice templates</a>	<a href="#">Referral Game Mechanics</a>	<a href="#">Resource blackout dates</a>	<a href="#">Referral settings</a>
<a href="#">Resource blackout dates</a>		<a href="#">Smart Grid Game Library</a>	
<a href="#">Smart Grid Game Library</a>		<a href="#">Library column names</a>	
<a href="#">Library column names</a>		<a href="#">Library actions</a>	
<a href="#">Library actions</a>		<a href="#">Smart Grid Game Designer</a>	
<a href="#">Smart Grid Game Designer</a>		<a href="#">Drafts</a>	
<a href="#">Drafts</a>		<a href="#">Designer actions</a>	
<a href="#">Designer actions</a>		<a href="#">Designer levels</a>	

Figure 3.2: Challenge Design Settings Page

Makahiki provides a Smart Grid Game Designer page for game designers to design the smart grid game. It provides a WYSIWYG interface to create educational content, design the grid layout and levels. Game designers use this page to construct the Smart Grid Game, by dragging and dropping the squares (game actions) into the grid with the intended layout. The interface is shown in Figure 3.3:

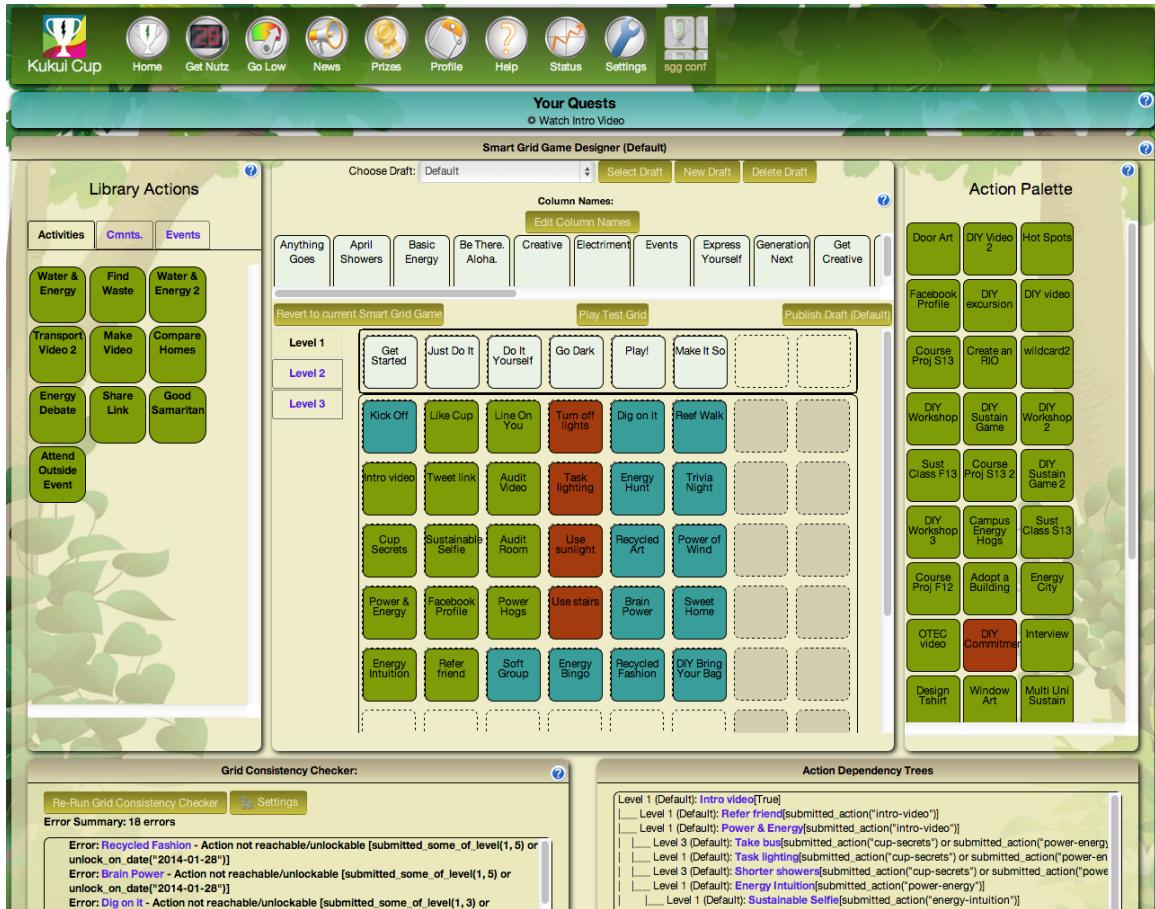


Figure 3.3: Smart Grid Game Designer Page

### 3.2.3 Player Interface

This section describes the web interface that is viewed by the players of the Makahiki system. Because Makahiki provides the functionality for a game designer to configure and customize the game he creates, the elements and looks in the player interfaces described here are also customizable.

#### 3.2.3.1 Landing Page

Figure 3.4 shows a sample landing page of Makahiki system.



Figure 3.4: Landing page

The landing page is the first page encountered by new player. So far, challenges built using Makahiki have a “closed” registration model; that is, the users of the system are known in advance and set up during the configuration process. Thus, the landing page has two buttons: one for users who live in a particular place and thus should have access to the system, and one for those who are

just visiting and would like to learn more about the system.

Most of the content on this page is configurable, including the University logo, the slogan, the text fields and button contents, and the sponsors.

Figure 3.5 shows another landing page that is customized to be used in the Kukui Cup at East West Center.

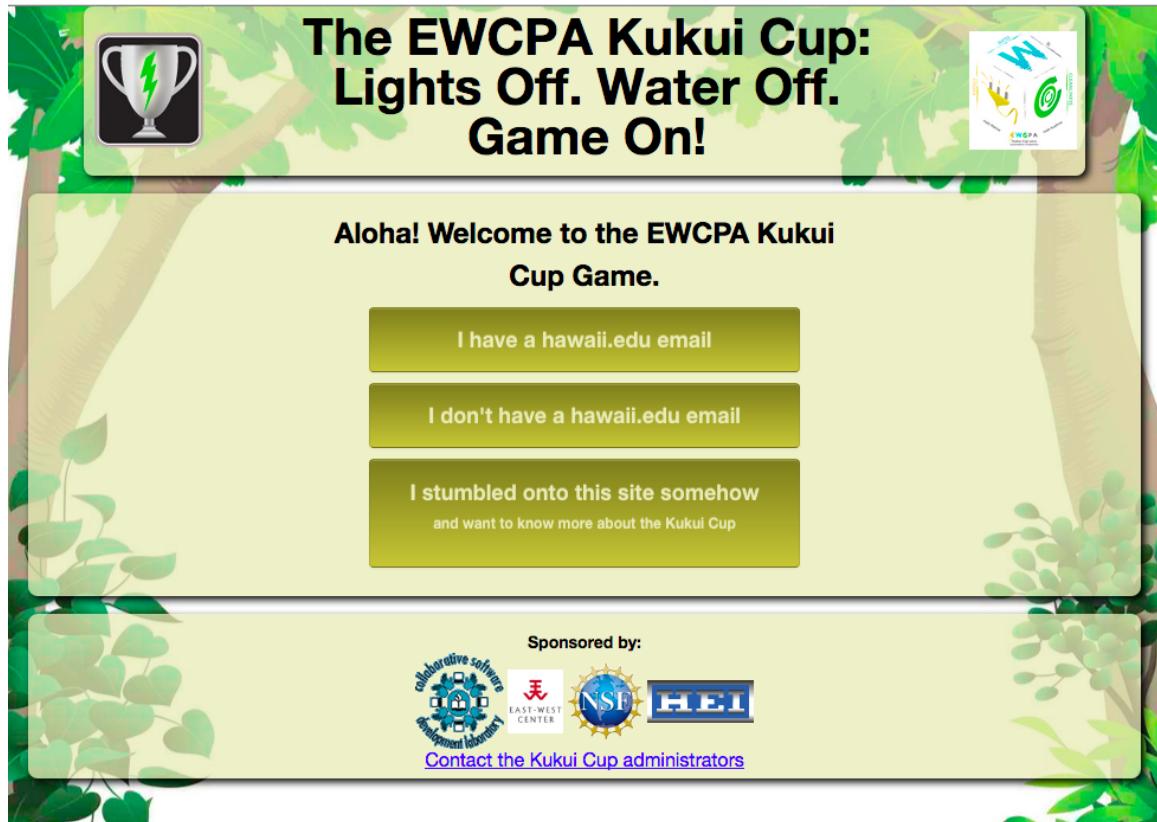


Figure 3.5: Landing page for East West Center Kukui Cup

### 3.2.3.2 Authentication Page

Users who click on the top button on the landing page are taken to an authentication page. Makahiki supports CAS (Central Authentication Service), LDAP, and Django internal authentication mechanisms. Different organizations can choose to configure the desired authentication method for their game. They can use any one or combination of the above methods. [Figure 3.6](#) shows the University of Hawaii Kukui Cup authentication screen that use the CAS authentication. [Figure 3.7](#) shows another authentication screen, the Django internal authentication, which is used in the East West Center Kukui Cup.

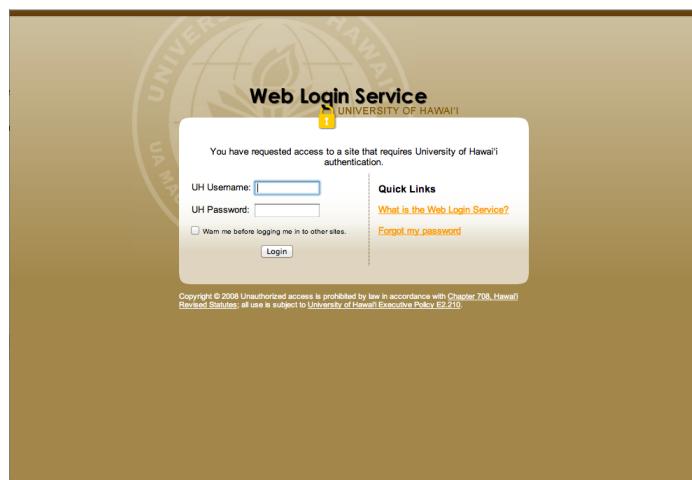


Figure 3.6: Authentication page for University of Hawaii Kukui Cup



Figure 3.7: Authentication page for East West Center Kukui Cup

### 3.2.3.3 Home Page

After logging in, the players are taken to the Home page, as shown in [Figure 3.8](#). The system sets a cookie when the player authenticates successfully, thus, after the first visit, the player will normally go directly to the home page when retrieving the challenge URL.



Figure 3.8: Home page

There are three UI components in the home pages that also appear on every page. They are:

- The “Info Bar” is a horizontal UI component at the top of every page. It provides status information about the challenge as well as a logout link. (The player will not normally need to logout unless they are accessing the site from a public computer.)
- The “Nav Bar” is a horizontal UI component below the Info Bar, which provides icons that

link to all of the top-level pages in the system. The set of pages in the system is configurable.

- The “Quest Bar” is a horizontal UI component below the Nav Bar. It provides “Quests” (explained in more detail below).

In the center of the home page are the clickable icons and description buttons that link to the main functional pages of the game. It provides a top level glance of the functionalities of the game. The set of functional pages is configurable using the game design admin interface. [Figure 3.9](#) shows another home page that is used in the Hawaii Pacific University (HPU) Kukui Cup. Note that there is no “Drop Down” function (for the water conservation game) in the HPU’s home page. There are two additional buttons of “Status” and “Settings” in HPU’s home page which are only available to the site administrators. The background and the color theme are also different.



Figure 3.9: Home page for Hawaii Pacific University Kukui Cup

The following sections describe in detail each of these top-level functional pages.

### 3.2.3.4 Get Nutz Page

The “Get Nutz” page provides the user interface to the primary sustainability education game, also known as the “Smart Grid Game”. Players gain points by clicking on cells in the Smart Grid UI widget, which takes them to the online or real-world educational actions such as activities, commitments and events.

**Figure 3.10** shows an example of the Get Nutz page (the name of this page and any other top-level page can be configured by the game designer).

The screenshot displays the 'Your Quests' section of the 'Get Nutz' page. At the top, there's a navigation bar with icons for Home, Get Nutz, Go Low, Drop Down, News, Prizes, Profile, Help, Status, and Settings. Below the navigation bar is a green header titled 'Your Quests' with three tabs: 'Learn Secrets of Kukui Cup Masters', 'Learn about energy', and 'Make a commitment'. The main content area is divided into several sections:

- The Smart Grid Game:** A 4x6 grid where players can click on cells to earn points. The grid contains numbers like 10, 20, 30, and some locked cells with padlocks. A legend below the grid defines symbols: green square for activity, red square for commitment, blue square for event, purple square for excursion, grey square for filler, orange circle for Submitted, green checkmark for Approved, and red circle with a minus sign for Rejected.
- Upcoming Events (Next 7 days):** A table listing five events with their dates and locations:
 

Event	Date	Location
Play outside the cafe (1)	Wed 04/01, 6:20 PM	Hale Aloha courtyard
Energy scavenger hunt	Wed 04/01, 6:20 PM	Lokelani rooftop
Play outside the cafe (2)	Fri 04/03, 6:20 PM	Hale Aloha courtyard
Flashmob design	Fri 04/03, 6:20 PM	Lehua rooftop
Kahuku Wind Farm	Sun 04/05, 6:20 PM	Hale Aloha courtyard
- Enter Attendance or Bonus Points Code:** A text input field with a 'Submit' button.
- Top Score Game (Individual, Round 1):** A scoreboard showing the top 10 individual scores:
 

Rank	Individual	Points
1	Player39 E	25
2	Player37 O	25
3	Player35 C	25
4	Player33 B	25
5	Player31 A	25
6	Player29 -E	25
7	Player27 -D	25
8	Player25 -C	25
9	Player23 -B	25
10	Player21 -A	25
- Active Participation Game (Overall):** A section showing participation levels:
 

Level	Participation
100% participation (10 points)	Be the first team to achieve this level
75% participation (5 points)	Be the first team to achieve this level
50% participation (5 points)	Be the first team to achieve this level
Below 50% participation	Lounge Lehua-A: 0% Lehua-B: 0%

Figure 3.10: Get Nutz Page

The page also provides widgets about upcoming events, a scoreboard showing top scoring leaders and a widget called “Active Participation Game.” The set of widgets appearing on this page is configurable. The game designer can use the game design admin interface to customize what widgets to include and where the widgets appear.

Figure 3.11 shows another example of the Get Nutz page, which displays a different layout of the Smart Grid Game, and a different position for the “Upcoming Events” widget. There is no “Active Participation Game” in this page.

The screenshot shows the Get Nutz page with a tropical theme. At the top, there are user profiles for 'pmj' and 'Ilima-E', and a footer for 'Kahuku Wind Farm' dated June 11, 2012, at 3:03 p.m. with a 'Logout' button and a 'Send Feedback' link.

**Smart Grid Game:** This section contains three tabs: Level 1, Level 2, and Level 3. The Level 1 tab is active, showing a grid of 12x12 cells. Some cells contain numbers (e.g., 10, 20, 30) while others have icons like locks or question marks. A legend at the bottom defines the colors: green for activity, red for commitment, blue for event, and purple for excursion.

**Your Quests:** A banner below the game lists three goals: "Learn Secrets of Kukui Cup Masters", "Learn about energy", and "Win the \$10 bonus!"

**Upcoming Events (Next 7 days):** This section displays a table of events with columns for Event, Date, and Location. The events include "Play outside the cafe (1)" on Thursday, June 07, at 3:03 PM in Hale Aloha courtyard, and "Energy scavenger hunt" on the same day at Lokelani rooftop.

Event	Date	Location
Play outside the cafe (1)	Thu 06/07, 3:03 PM	Hale Aloha courtyard
Energy scavenger hunt	Thu 06/07, 3:03 PM	Lokelani rooftop
Recycled fashion design	Fri 06/08, 3:03 PM	Ilima rooftop
Play outside the cafe (2)	Sat 06/09, 3:03 PM	Hale Aloha courtyard
Flashmob design	Sat 06/09, 3:03 PM	Lehua rooftop
Kahuku Wind Farm	Mon 06/11, 3:03 PM	Hale Aloha courtyard
Sustainable and Organic Farming	Mon 06/11, 3:03 PM	Hale Aloha courtyard
Pedalpalooza	Tue 06/12, 3:03 PM	Hale Aloha courtyard

**Round 1 Point Scoreboard:** This section shows a table of individual scores for Ilima-E. The top 10 individuals are listed with their names and points.

Rank	Individual In Ilima-E	Points
1	pmj	25
2	testuser4-Ilma-E	25
3	testuser3-Ilma-E	25
4	testuser2-Ilma-E	25
5	testuser1-Ilma-E	25
6	testuser4-Lokelani-D	25
7	testuser3-Lokelani-D	25
8	testuser2-Lokelani-D	25
9	testuser1-Lokelani-D	25
10	testuser4-Lokelani-D	25

A message at the bottom right of the scoreboard says "#1 at end of round wins a prize!"

Figure 3.11: Another Get Nutz Page

Clicking on the link in the middle of the smart grid game takes the player to a page explaining that educational action. **Figure 3.12** shows an example of the action page.

The screenshot shows a game interface with a green header bar. On the left is a user icon labeled 'pmj'. In the center, it says 'Round 1: 25 points #1/102 overall.' and 'Overall: 125 points. #1 out of 20'. On the right, it shows 'Flashmob design' and the date 'June 9, 2012, 3:03 p.m.' with 'Logout' and 'Send Feedback' buttons.

The main content area has a green header titled 'Your Quests' with three items: 'Learn Secrets of Kukui Cup Masters', 'Learn about energy', and 'Win the \$10 bonus!'. Below this, there are two panels:

- Excursion : Wind Farm** (Left Panel):
  - This excursion is worth **30 points**.
  - When :** June 11, 2012, 3:03 p.m.
  - Where :** Hale Aloha courtyard
  - Estimated duration :** 4 hours
  - Space(s) still available :** 25
  - Social Bonus :** 10 points.
  - 0 people in the competition have signed up.
  - 0 people in your lounge have signed up.
  - Get a text or email reminder [Remind Me](#)
- About the excursion** (Right Panel):
  - Kahuku Wind Farm**: An image of a tall red lattice wind turbine tower with a white nacelle and blades.
  - Text: "Want to see sky farmers harvesting the winds? Come with us to Kahuku to see firsthand how energy is plucked from the sky and generated for our use by First Wind's turbines."
  - Text: "The wind farm staff requests that everyone wear long pants and closed toe shoes. We hope to stop in Kahuku for lunch, so bring some money."
  - Text: "You need to register for this free event to reserve your seat on the bus by clicking the I want to sign up button below."
  - Text: "Meet in the Hale Aloha courtyard, from there we'll get on the bus. Don't be late!"
  - [I want to sign up](#)

Figure 3.12: Action Page

### 3.2.3.5 Go Low Page

The “Go Low” page provides the user interface to two “energy” games, as shown in [Figure 3.13](#):

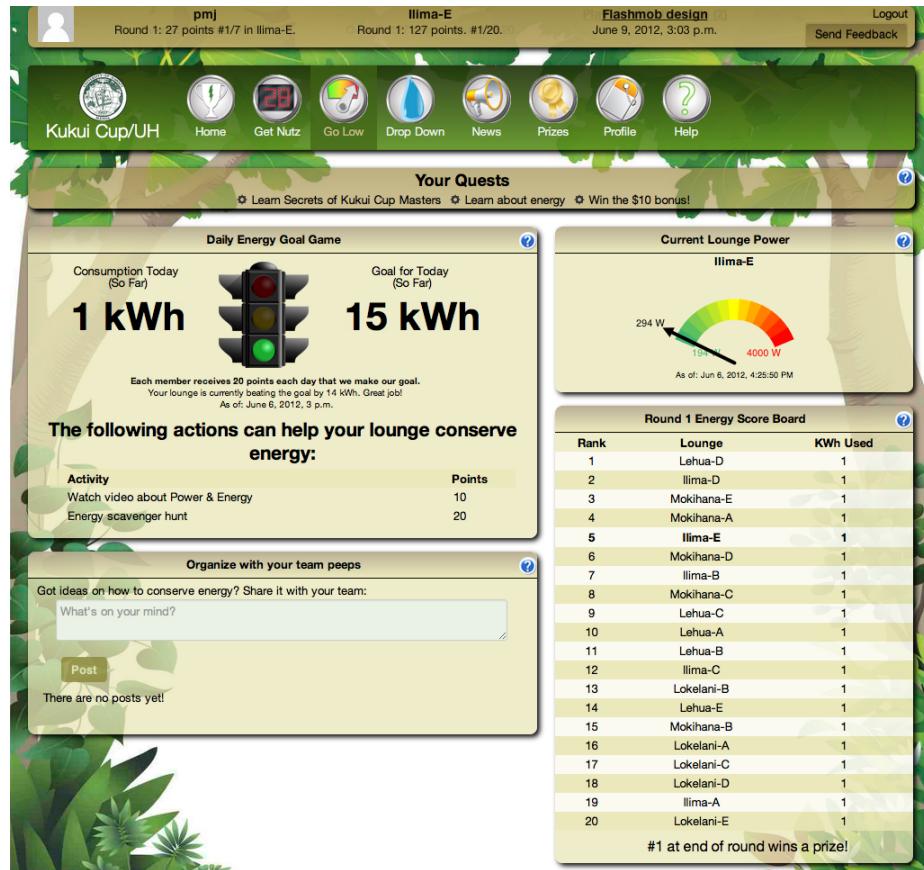


Figure 3.13: Go Low Page

On the left side, the “Daily Energy Goal Game” is designed to incentivize players to reduce their energy usage by awarding them points if they can reduce their team’s energy by a certain percentage below a baseline value. The stoplight visualization tells them whether or not they are currently on track to make the reduction goal.

On the right side, the “Current Power” visualization helps players to see what their current power consumption is in near real-time (typically every 10-15 seconds.)

The page also enables team members to communicate via a shared chat window, and provides a scoreboard widget showing leaders in energy conservation.

### 3.2.3.6 Drop Down Page

The “Drop Down” page provides the user interface to the “water” games, as shown in [Figure 3.14](#):

The screenshot shows a competition interface for the EWC Kukui Cup. At the top, two dormitories are listed: Hale Kuahine (1821 points, #1 of 133 overall) and Hale Manoa (11394 points, #2 of 2). The current round is Round 1, ending on March 31, 2015, at 11:59 p.m. Navigation links include Home, Get Nutz, Go Low, Drop Down, News, Profile, and Help.

**Daily Water Goal Game:**

Mon	Tue	Wed	Thu	Fri	Sat	Sun
9/3 X usage: 1602 gal	9/4 X usage: 1602 gal	9/5 X usage: 1581 gal	9/6 X usage: 1571 gal	9/7 X usage: 1558 gal	9/8 X usage: 1629 gal	9/9 X usage: 1610 gal
9/10 ✓ usage: 1528 gal	9/11 ✓ usage: 1518 gal	9/12 ✓ usage: 1506 gal	9/13 ✓ usage: 1496 gal	9/14 ✓ usage: 1486 gal	9/15 II goal: 1505 gal (reduce 5%)	9/16 II goal: 1510 gal (reduce 4%)
9/17 II goal: 1502 gal (reduce 4%)	9/18 II goal: 1493 gal (reduce 4%)	9/19 II goal: 1481 gal (reduce 4%)	9/20 II goal: 1472 gal (reduce 4%)	9/21 II	9/22 II	9/23 II

**Water Goal Game Scoreboard (Overall):**

Rank	Student dormitory	Days met goal	Avg reduction
1	Hale Kuahine	6	1%
2	Hale Manoa	4	7.5%

Figure 3.14: Drop Down Page

On the left side, the “Daily Water Goal Game” is designed to incentivize players to reduce their water usage by awarding them points if they can reduce their team’s water usage by a certain percentage below a baseline value. The calendar visualization tells them whether or not they made the goal for the individual day during the current round of the competition.

On the right side, the “Water Goal Game Scoreboard” shows the leaders in water conservation.

### 3.2.3.7 News Page

The “News” page provides information about the state of the challenge and the team of which this player is a member, as shown in [Figure 3.15](#):

Event	Date	Location
Play outside the cafe (1)	Thu 06/07, 3:03 PM	Hale Aloha courtyard
<a href="#">Energy scavenger hunt</a>	Thu 06/07, 3:03 PM	Lokelani rooftop
Recycled fashion design	Fri 06/08, 3:03 PM	Ilma rooftop
Play outside the cafe (2)	Sat 06/09, 3:03 PM	Hale Aloha courtyard
Flashmob design	Sat 06/09, 3:03 PM	Lehua rooftop
Kahuku Wind Farm	Mon 06/11, 3:03 PM	Hale Aloha courtyard
Sustainable and Organic Farming	Mon 06/11, 3:03 PM	Hale Aloha courtyard
Pedalpalooza	Tue 06/12, 3:03 PM	Hale Aloha courtyard

Figure 3.15: News Page

Widgets such as “Lounge Members,” “Most Popular Excursion,” “My Public Commitments,” etc. all provide a sense for the state of the competition and encourage players to participate by learning about what others members are doing.

### 3.2.3.8 Prizes Page

The “Prizes” page provides access to two games: the “Top Scorer” game and the “Raffle” game, as shown in [Figure 3.16](#):

The screenshot shows the Prizes Page of a competition website. At the top, there are user profiles for 'pmj' (Overall: 27 points #1/102) and 'Ilma-E' (Overall: 127 points. #1 out of 20). The date is June 7, 2012, 3:03 p.m. The top navigation bar includes links for Home, Get Nutz, Go Low, Drop Down, News, Prizes, Profile, and Help.

**Your Quests**

- Learn Secrets of Kukui Cup Masters
- Learn about energy
- Win the \$10 bonus!

**Prizes and Prize Winners**

Overall Prizes	
Prize	Criteria
iPad	iPad 2 for the individual with the greatest number of points across all four towers at the end of the Quest for the Kukui Cup. Current leader: pmj
Bookstore Certificate	\$10 Bookstore Gift Certificate for the individual with the most points on each lounge at the end of the Quest for the Kukui Cup. Current leader: pmj
Energy Pizza Party	Pizza party for the lounge with the lowest energy consumption over the entire Quest for the Kukui Cup. Current leader:
Point Pizza Party	Pizza party for the lounge whose members earned the greatest number of points over the entire Quest for the Kukui Cup. Current leader: Ilma-E

**Round 1 Raffle Game**

Your total raffle tickets: 1 Allocated right now: 0 Available: 1					
Prize	Value	Your Tickets	Total Tickets	Current Odds	Change Ticket Allocation
Skateboard	\$180.00	0	0	0.0%	+1 -1
Energy Meter (2)	\$30.00	0	0	0.0%	+1 -1
Energy Meter (1)	\$30.00	0	0	0.0%	+1 -1
Energy Meter (3)	\$30.00	0	0	0.0%	+1 -1
JJ tshirt (M)	\$20.00	0	0	0.0%	+1 -1
JJ tshirt (L)	\$20.00	0	0	0.0%	+1 -1
JJ tshirt (XL)	\$20.00	0	0	0.0%	+1 -1
JJ tshirt (S)	\$20.00	0	0	0.0%	+1 -1
Rash Guard	\$18.00	0	0	0.0%	+1 -1
BM Cami	\$15.00	0	0	0.0%	+1 -1
Board Shorts	\$15.00	0	0	0.0%	+1 -1
Belly Dance (2)	\$15.00	0	0	0.0%	+1 -1
Backpack	\$15.00	0	0	0.0%	+1 -1

Figure 3.16: Prizes Page

The Top Scorer game, illustrated by the widget on the left, shows the prizes that can be won by top scorers in the competition.

The Raffle Game provides an alternative route to win a prize. Here, players earn in-game raffle tickets based upon their point score that can be allocated to any of a collection of raffle prizes. The odds of winning are based upon the percentage of their tickets allocated to the prize, which is picked at random at the end of a round by administrators.

The Raffle Game provides an incentive for players to do activities and earn points even if they do not stand a chance of winning one of the Top Scorer prizes.

### 3.2.3.9 Profile Page

The “Profile” page provides access to profile information for this player, as shown in [Figure 3.17](#):

The screenshot shows the Makahiki Profile Page for user pmj. At the top, there are three tabs: "pmj" (Overall: 27 points #1/102), "Ilima-E" (Overall: 127 points. #1 out of 20), and "Energy scavenger hunt" (June 7, 2012, 3:03 p.m.). Below the tabs is a navigation bar with links: Kukui Cup/UH, Home, Get Nutz, Go Low, Drop Down, News, Prizes, Profile, and Help. The main content area is titled "Your Quests" and lists three items: "Learn Secrets of Kukui Cup Masters", "Learn about energy", and "Win the \$10 bonus!". The "My Info" section contains fields for Display name (pmj), Picture (placeholder), and Theme (Forest). The "Reminder Settings" section includes Email (johnson@hawaii.edu) and Text (808-888-8888 / Verizon). The "My Badges" section indicates no badges yet. The "My Public Commitments" section states no commitments. The "My Achievements" section is divided into "Current and Pending" and "Previously Awarded". "Current and Pending" shows one task: "Excursion: Wind Farm" (Signed up). "Previously Awarded" shows three tasks with points: "Excursion: Kahuku Wind Farm (Sign up)" (2 points), "Activity: Watch introduction video" (20 points), and "Set up profile" (5 points).

Figure 3.17: Profile Page

The user can set their display name, their picture, and how they wish to be contacted for reminders. It also shows information about their badges and a complete record of how they earned all of the points in the game.

The profile page also allows them to change the theme associated with the site. A variety of themes are available. In this configuration, the default theme is “Forest”, but users can go to the Profile page to set a different theme according to their own personal preferences. Makahiki bundles 8 different themes that can be selected from the theme dropdown menu in the Profile page. Makahiki also provides a development interface for UI developers to create more themes and include new themes into the system.

Figure 3.18 and Figure 3.19 shows two other themes that a player can select to customize the theme for his game interface.



Figure 3.18: Profile Page with the Google theme

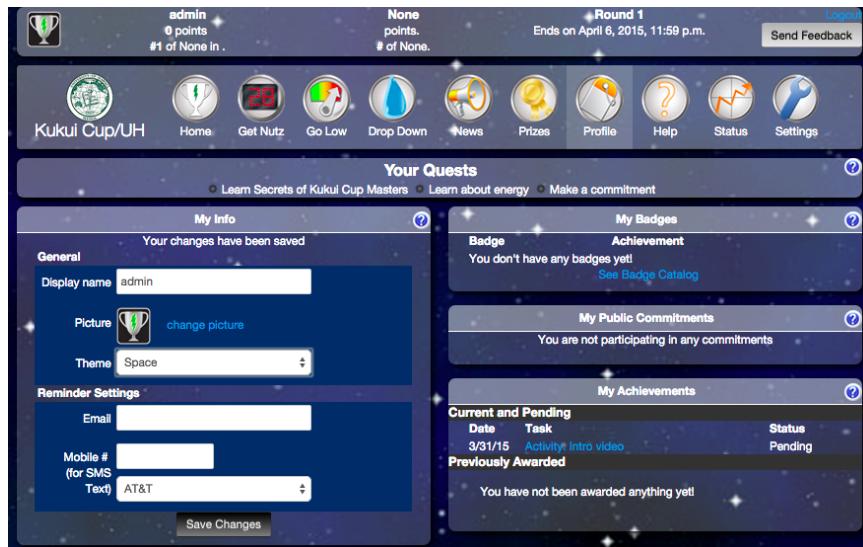


Figure 3.19: Profile Page with the Space theme

### 3.2.3.10 Help Page

The final page available to players is the Help page, which simply provides access to explanatory material about the system, as shown in [Figure 3.20](#):

The screenshot shows the Help page of the Kukui Cup/UH competition. At the top, there are three tabs: "pmj" (Round 1: 27 points #1/7 in Ilima-E), "Ilima-E" (Round 1: 127 points, #1/20), and "Energy scavenger hunt" (June 7, 2012, 3:03 p.m.). The "Logout" button is also visible. Below the tabs is a navigation bar with icons for Home, Get Nutz, Go Low, Drop Down, News, Prizes, Profile, and Help. The main content area is titled "Your Quests" and includes links to "Learn Secrets of Kukui Cup Masters", "Learn about energy", and "Win the \$10 bonus!".  
  
The central feature is a video player titled "Introduction to the Kukui Cup Grid Game". The video shows a 4x7 grid of numbered squares (10, ?, 10, ?, 10, ?, 10) and question marks. Below the video player is a "Frequently Asked Questions" section with links to topics like "How is the energy goal picked?", "What is plug load?", etc.  
  
To the right of the video player is a "Rules of the competition" section containing links to various rules and guidelines. At the bottom right is a "Ask an Admin" form with a text input field and a "Send" button.

Figure 3.20: Help Page

### 3.2.4 Game Manager Admin Interface

A game manager uses the “Challenge Management” admin interface to manage the game. He can view and search the system logs to identify errors, view the goal status for energy and water usage reduction, monitor the wall postings from the players, etc.

In order to manage the individual games in the challenge, a game manager uses the game management section for each game. For example, in the Energy and Water game section, he can view the energy and water usage in the case of the usage data is automatically retrieved from smart meters. In the case of not using smart meters, a game manager can use this interface to manually enter the usage data.

One important task to manage the Smart grid game is to approve the submissions from players. The game manager can use the “Action Submissions” link in the smart grid game management section to view the submissions and approve them, as well as generate the bonus point coupons to be handed out to players.

Figure 3.21 shows the challenge management interface.

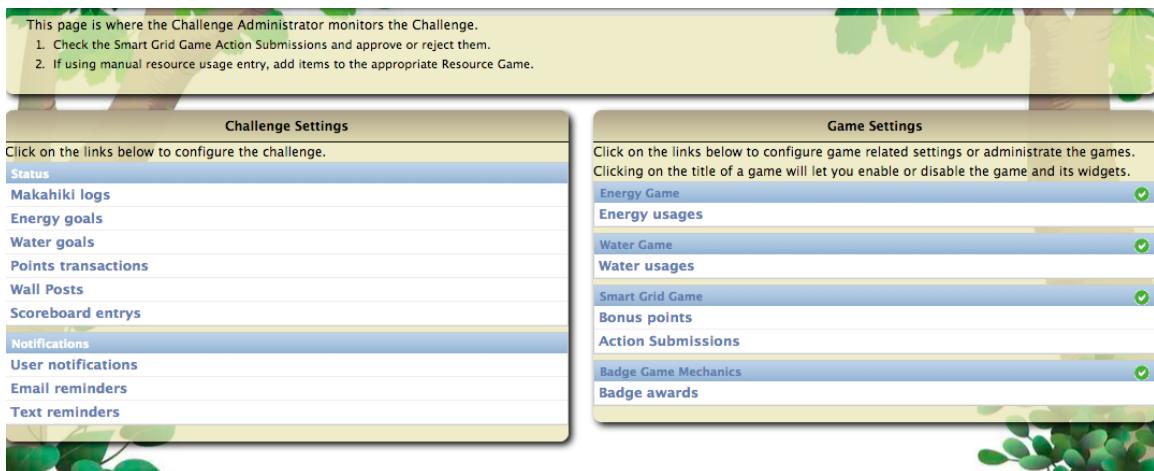


Figure 3.21: Challenge Management Settings Page

### 3.2.5 Game Developer Admin Interface

Game developers use the “Developer” admin interface to debug the system, add and configure new pages and games. [Figure 3.22](#) shows the developer interface.

This page is where the Developer can edit the settings. The developer has complete control over all the settings.

Challenge Settings		Game Settings	
Click on the links below to configure the challenge.			
<b>Challenge</b>		<b>Game Settings</b>	
<a href="#">Global settings</a>		<a href="#">Energy Game</a> <input checked="" type="checkbox"/>	
<a href="#">Round settings</a>		<a href="#">Energy usages</a>	
<a href="#">Sponsors</a>		<a href="#">Energy goal settings</a>	
<a href="#">Point rubrics</a>		<a href="#">Energy goals</a>	
<a href="#">Help topics</a>		<a href="#">Energy daily baselines</a>	
<b>Players</b>		<a href="#">Energy hourly baselines</a>	
<a href="#">Groups</a>		<a href="#">Water Game</a> <input checked="" type="checkbox"/>	
<a href="#">Teams</a>		<a href="#">Water usages</a>	
<a href="#">Users</a>		<a href="#">Water goal settings</a>	
<a href="#">Avatars</a>		<a href="#">Water goals</a>	
<b>Status</b>		<a href="#">Water daily baselines</a>	
<a href="#">Makahiki logs</a>		<a href="#">Water hourly baselines</a>	
<a href="#">Points transactions</a>		<a href="#">Smart Grid Game</a> <input checked="" type="checkbox"/>	
<a href="#">Scoreboard entries</a>		<a href="#">Action feedbacks</a>	
<a href="#">Wall Posts</a>		<a href="#">Bonus points</a>	
<a href="#">Notice templates</a>		<a href="#">Actions</a>	
<a href="#">User notifications</a>		<a href="#">Column names</a>	
<a href="#">Daily status</a>		<a href="#">Levels</a>	
<a href="#">Email reminders</a>		<a href="#">Action Submissions</a>	
<a href="#">Text reminders</a>		<a href="#">Column grids</a>	
<b>Games</b>		<a href="#">Grids</a>	
<a href="#">Page infos</a>		<a href="#">Top Score Game</a> <input checked="" type="checkbox"/>	
<a href="#">Game infos</a>		<a href="#">Prizes</a>	
<a href="#">Smart Grid Game Library</a>		<a href="#">Raffle Game</a> <input checked="" type="checkbox"/>	
<a href="#">Library column names</a>		<a href="#">Raffle prizes</a>	
<a href="#">Library actions</a>		<a href="#">Participation Game</a> <input checked="" type="checkbox"/>	
<b>Smart Grid Game Designer</b>		<a href="#">Participation settings</a>	
<a href="#">Drafts</a>		<a href="#">Team participations</a>	
<a href="#">Designer actions</a>		<a href="#">Quest Game Mechanics</a> <input checked="" type="checkbox"/>	
<a href="#">Designer levels</a>		<a href="#">Quests</a>	

Figure 3.22: Developer Settings Page

### 3.2.6 Game Analytics

The “Status” page is restricted to administrators such as game designers and game managers. It provides the game analytics which displays numbers of analytics widgets about the state of an ongoing challenge and the results of the challenge. Figure 3.23 shows a screenshot of part of the Status page:

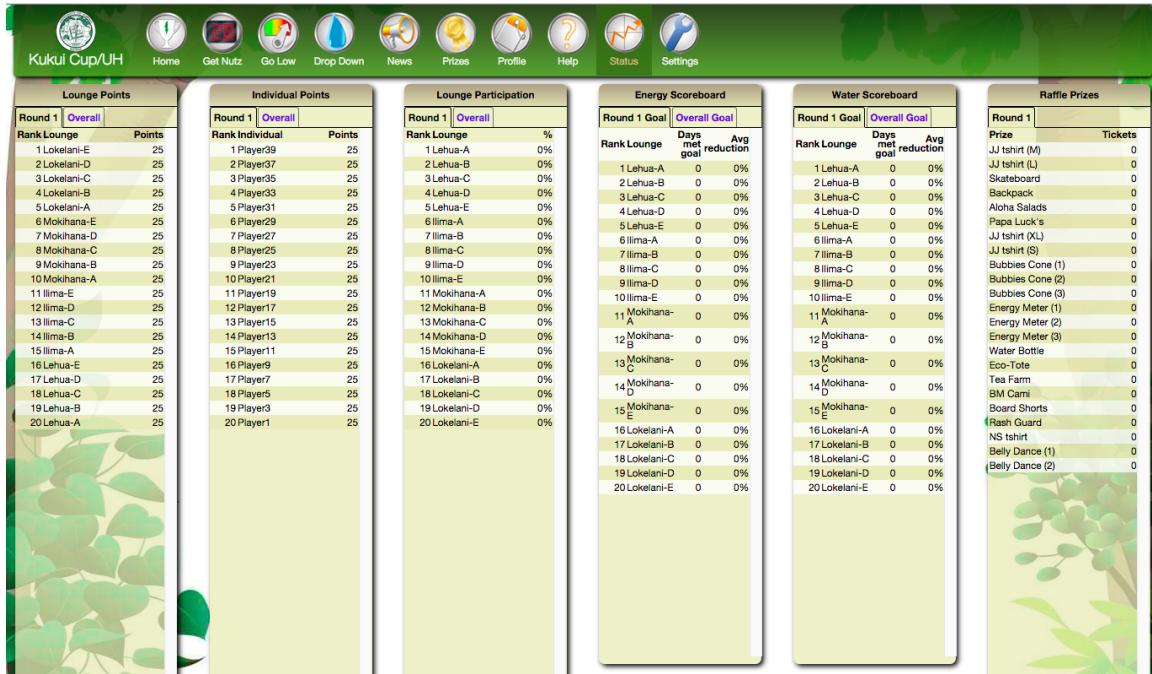


Figure 3.23: Status Page

The Status page is designed to help administrators monitor the progress of their challenge, detect problems with the system and/or state of play, and intercede to correct them in a timely manner.

### 3.3 System Design

This section describes the design of Makahiki. It first illustrates the architecture of Makahiki, followed by the descriptions of design features that makes Makahiki an innovative serious game framework.

#### 3.3.1 Architecture

Figure 3.24 illustrates the overall architecture of Makahiki. The core component of Makahiki is a configurable game engine that can be customized to the needs of different organizations. It includes two libraries of games and game mechanics. These libraries consist of a set of pre-built “widgets.” By selecting and configuring these game and game mechanics widgets, an organization can create a customized energy and water challenge (or serious game) in which players compete both individually and in teams, earn points by reducing their energy consumption as well as by learning about energy concepts in general, and are provided with incentives such as top score prizes and raffle prizes.

Makahiki interfaces with the outside environment in three different ways.

First, the top side of the architecture diagram shows that Makahiki has two primary user interfaces: one for the players of the serious game, who directly interact with the game and game mechanics widgets; the other for the administrators of the system, who configure the system and monitor the real-time game analytics.

Second, the right side of the diagram illustrates that Makahiki must obtain real-world environmental data as the challenge progresses in order to provide feedback to users about the impact of their actions. In some cases, environmental data can be input automatically into the system through a combination of “smart” meters and additional services (such as the WattDepot system for energy data collection, storage, and analysis). If that is not possible, then manual meters can be read by administrators on a regular (typically daily) basis and input into Makahiki using the administrator interface.

Third, the bottom side illustrates that Makahiki stores its data in a database repository (currently

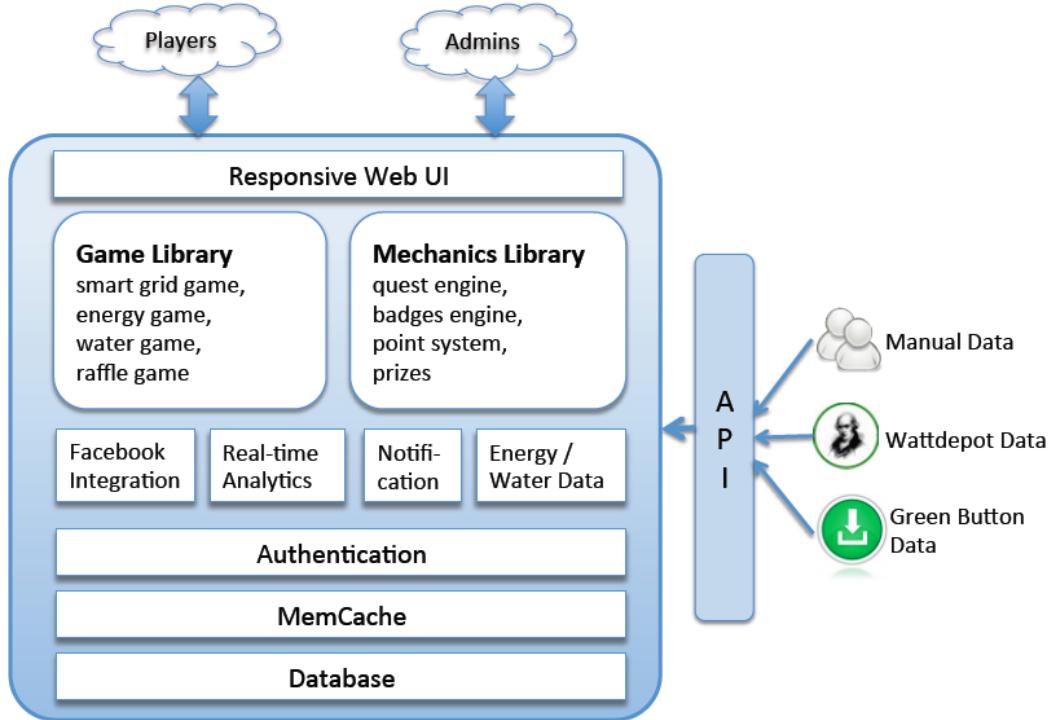


Figure 3.24: Architecture of Makahiki

PostgreSQL). To reduce database access and improve performance, Makahiki provides support for caching (currently memcached).

[Figure 3.25](#) illustrates the internal architecture of Makahiki. It provides a perspective on Makahiki's internal architecture in terms of three kinds of “components”: Django-related infrastructure, Widgets, and Managers.

Managers are modules that provide Makahiki capabilities that do not involve a (player) user interface. They might provide interaction with administrators via the Django admin interface. Managers can implement game mechanic data structures (such as scores, players, and teams) or more generic web service functions (transactions, authorization, etc.)

Widgets are modules that provide Makahiki capabilities that do include a player user interface. Widgets can be roughly characterized in three ways. “Info widgets” provide state information about the challenge to players but little in the way of interaction. “Mechanics widgets” provide game elements such as Quests and Badges. The third category, “game widgets”, refer to full-fledged

## Makahiki

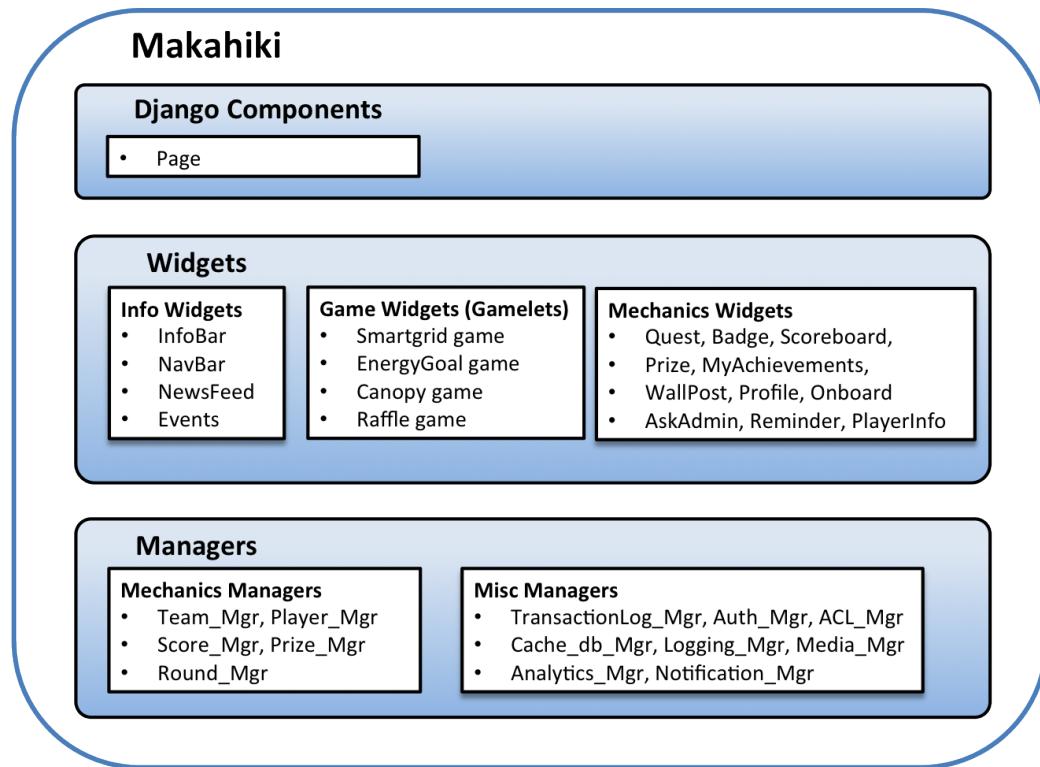


Figure 3.25: Internal Architecture of Makahiki

Manager	Module Name	Functionality
Authentication Manager	auth_mngr	Provides authentication services for Makahiki including administrative logins and CAS authentication
Cache Manager	cache_mngr	Manages caching data structures
Challenge Manager	challenge_mngr	Maintains state information about an entire Challenge, including what widgets are enabled, the round information
Log Manager	log_mngr	Provides logging services to track the actions of logged in users
Player Manager	player_mngr	Supports definition and processing of players
Resource Manager	resource_mngr	Provides supports for resource management, such as updating energy or water data, checking daily goals
Score Manager	score_mngr	Manages score keeping, scoreboard calculation
Team Manager	team_mngr	Support definition and management of teams

Table 3.2: Makahiki Internal Manager Modules

interactive games.

### 3.3.2 A Library of Configurable Games and Mechanics

Makahiki builds in a set of configurable games and mechanics that can be turned on or off, or customized by the game designers to the needs of different organizations. The rest of this section describes these games and game mechanics and how to configure them.

#### 3.3.2.1 Selecting a Game or Game Mechanics from the Library

A challenge designer can decide which of the built-in game and mechanics will appear in a sustainability challenge for his organization. The challenge design setting page includes a widget called “Game Settings”, which lists the set of games available and whether or not they are currently enabled, as is shown in [Figure 3.26](#).

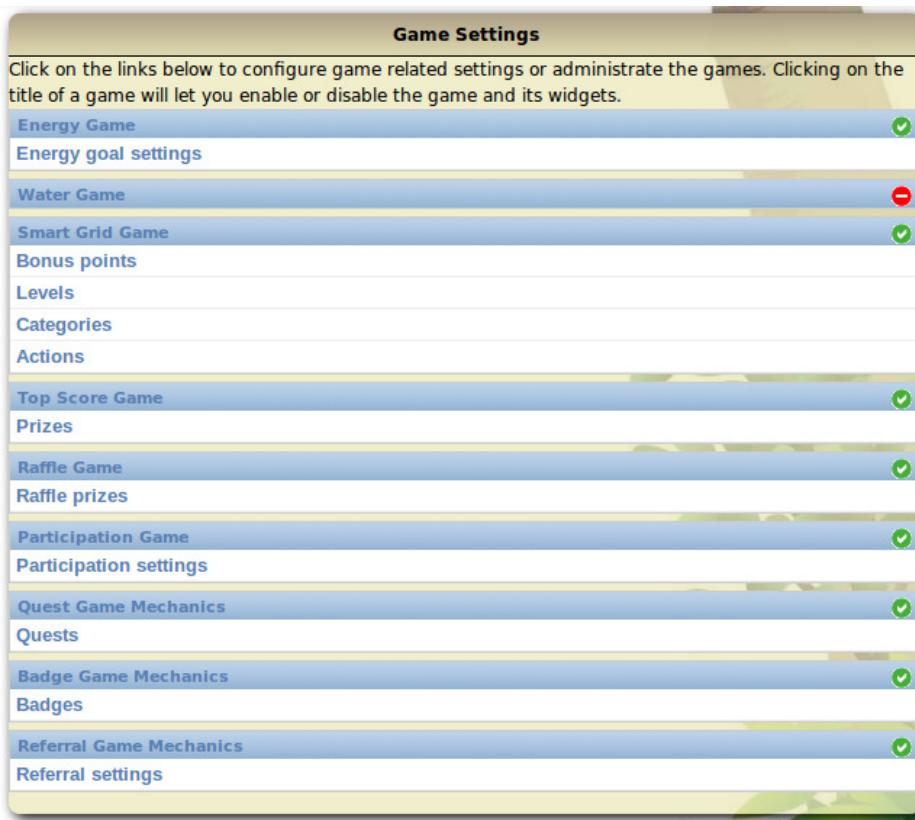


Figure 3.26: Selecting a Game from Game Settings

The small green and red icons on the right side indicate whether a game is currently enabled for the challenge. In this case, all of the games but one (Water Game) are enabled. Clicking on the title

of a game will let you enable or disable the game and its widgets. After clicking on the title link in the Game Admin widget, a page similar to the [Figure 3.27](#) will appear:

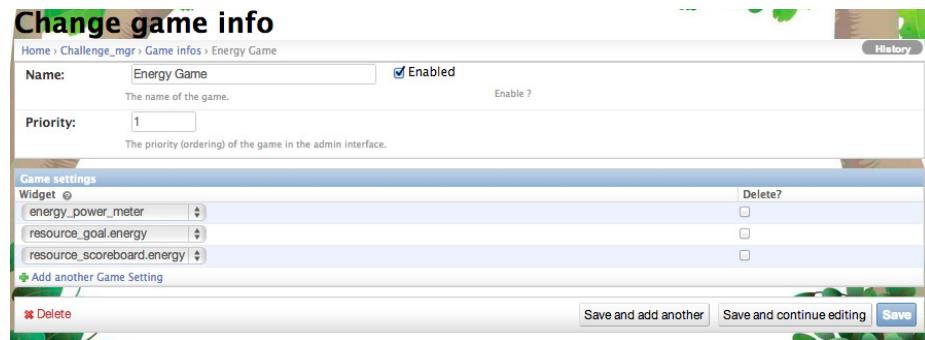


Figure 3.27: Change a Game's Setting

You can check or uncheck the “enabled” checkbox to enable or disable the game.

A game’s UI is represented by a set of widgets which is visible in the game website. The widgets belonging to a game are listed under the “Game Settings” section, as shown in the above screen shot. If you disable the game, all the widgets belonging to this game will be hidden in the web page.

Makahiki currently allows you to create a challenge out of the following games and game mechanics:

- *Energy Game*. This game awards points to players depending upon their ability to lower their energy consumption.
- *Water Game*. This game awards points to players depending upon their ability to lower their water consumption.
- *Smart Grid Game*. This game is the principle interface to the educational component of Makahiki. The SGG awards points to players for successfully completing activities, commitments, excursions, and events.
- *Top Score Game*. This game awards prizes to players and teams for earning the highest number of points during a round.
- *Raffle Game*. This game awards prizes to players if they have allocated their raffle tickets to

a particular raffle prize, and that raffle ticket was randomly selected by the system at the end of a round.

- *Participation Game.* This game awards points to players if they can successfully get a certain percentage of their team members to participate in the challenge.
- *Quest Game Mechanics.* This game mechanic provides a way for players to learn about features of the challenge by guiding them through Quests.
- *Badge Game Mechanics.* This game mechanic provides a way for players to earn badges for playing the game in a variety of ways.
- *Referral Game Mechanics.* This game mechanic provides a way for players to earn points by getting other people to participate in the challenge.

The following sections explain the design and configuration of each game and game mechanics.

### 3.3.2.2 Energy and Water Game

A fundamental requirement for enabling more active participation in sustainability behavior is feedback regarding their resource such as energy and water usage. The Energy and Water game in Makahiki are implemented as the Daily Resource Goal Game, which includes Daily Energy Goal Game and Daily Water Goal Game.

The Daily Energy Goal Game widget provides a way for players to see the outcome of the energy reduction behavior, and to make it a game by earning points from their behavior. By reducing their teams' daily energy consumption from a baseline by a set percentage, the players in the team will all earn a configurable amount of points. This baseline is calculated using historical data and recalculated dynamically throughout the competition. Both the baseline data and the current consumption is typically provided by API calls from Makahiki to an underlying WattDepot server.

[Figure 3.28](#) illustrates this widget.

As you can see, this interface uses a stoplight metaphor to show at a glance whether or not the team is making the goal. In this case, the stoplight is green, indicating they are currently below the goal.

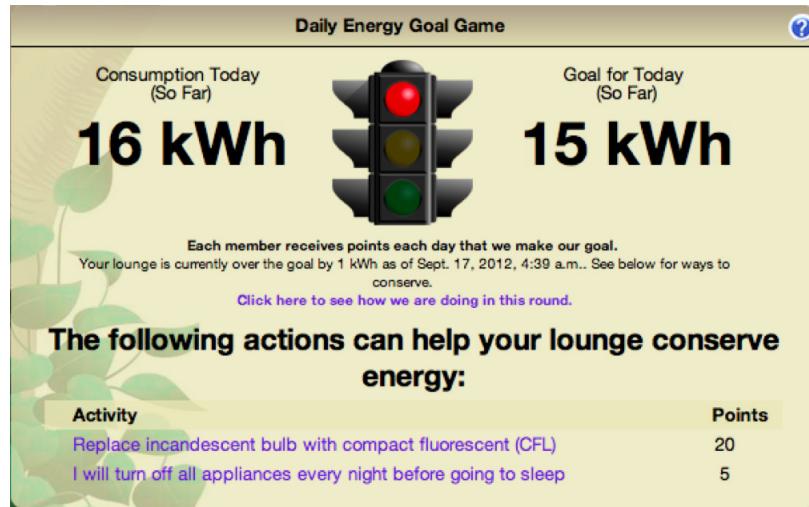


Figure 3.28: Daily Energy Goal Game widget

The goal for each team is typically a percent reduction from their baseline usage. When a player goes to the “Go Low” page of Makahiki, they can view their team’s current progress toward their daily energy goal. Near the end of the day, Makahiki checks the energy data from Wattdepot to see if a floor reached their goal. If the floor did reach their goal, each member of the floor that is participating in the game receives points. The energy goal game provides a link between the energy conservation competition and the point competition.

While the stoplight visualization provides good feedback to a team regarding their current progress toward making the current day’s goal, we have found additional perspectives to also be useful.

One useful perspective to a team is a realtime power meter visualization that shows the current power usage of a team, as shown in [Figure 3.29](#).

This visualization displays the realtime power consumption which updates at a specified interval. This gives players a sense of energy consumption at the moment. For example, someone turns on a high power microwave, they might see a spike in the realtime power meter reflecting the power usage at that moment.

Another useful perspective to a team is a historical, calendar-based visualization that shows the results of the energy goal game for each day of the current round, as shown in [Figure 3.30](#).

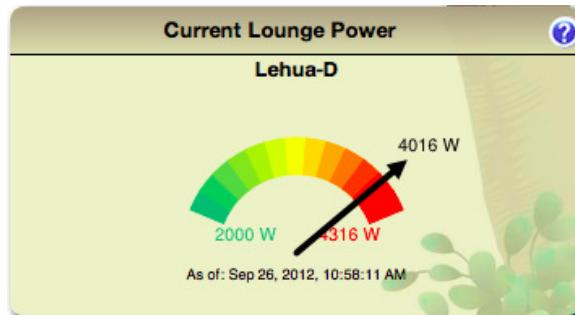


Figure 3.29: Power Meter widget



Figure 3.30: Daily Energy Goal Game Calendar widget

This visualization is useful for helping teams to see if there are patterns to their ability to make their goal. The above display shows that they have been making their goal more regularly in the recent past, indicating perhaps that they have identified a useful strategy for conservation.

Yet another perspective of the energy consumption is illustrated in Figure 3.31. It is the Daily Energy Goal Game Scoreboard, which shows how teams are faring relative to each other. It can

incentivize teams to conserve not only to earn points, but also to do better than other teams with respect to energy consumption. The scoreboard shows that the number of times that a team makes their daily energy goal and the average reduction percentage.

Energy Goal Game Scoreboard (Round 3)			?
Rank	Lounge	Days met goal	Avg reduction
1	Lokelani-E	20	8.6%
2	Mokihana-B	20	8%
3	Lokelani-D	19	5.6%
4	Lehua-B	18	6.2%
5	Mokihana-E	17	5%
6	Ilima-B	17	4.1%
7	Mokihana-C	15	2.2%
8	Lokelani-A	14	5.7%
9	Ilima-E	14	4.4%
10	Lehua-D	13	6%
11	Ilima-A	13	5.6%
12	Lokelani-B	13	4.8%
13	Ilima-C	13	3.6%
14	Mokihana-A	13	2.8%
15	Lehua-C	12	9.3%
16	Ilima-D	12	6.8%
17	Mokihana-D	12	6.8%
18	Lokelani-C	11	5.3%
19	Lehua-A	10	11.3%
20	Lehua-E	8	11.8%

1 2 3 4 5



Figure 3.31: Daily Energy Goal Game Scoreboard

The Daily Water Goal Game is similar to the Energy Goal game, with the only difference being the data, and whether the data is automatically or manually collected. With the installation of smart water meters and the availability of an automatic data collection system, all the visualizations discussed above will be available to the Water game. Otherwise, the water consumption data will have to be manually input daily. In this case, the hourly Water Goal Game widget and the near realtime Power Meter widget will not be available to the players, while the daily water scoreboard widget and calendar view widget will be available.

### 3.3.2.3 Smart Grid Game

Smart Grid Game (SGG) is Makahiki's approach to support "gamified" delivery of educational experiences. It is the primary place players go to learn about sustainability issues and earn points. Educational actions are organized into a grid of squares (hence the name "Smart Grid") and organized by category columns and levels. Players use its grid interface to discover "actions" they can perform. Successful completion of an action earns the player a variable number of points depending upon the difficulty of the action, and can potentially "unlock" additional actions and higher levels in the SGG. [Figure 3.32](#) shows a typical Smart Grid Game interface for players:

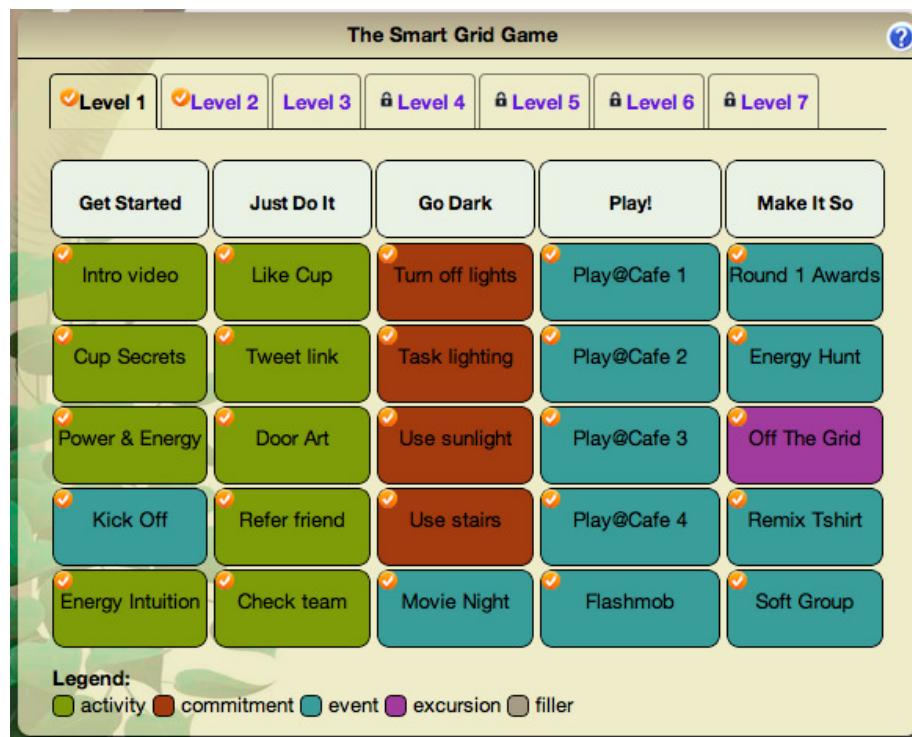


Figure 3.32: Smart Grid Game widget

There are three types of actions in the grid:

*Activities* are the most basic action available in the Smart Grid. In order to get points for an activity, a player must input a response to the system, which is reviewed and approved or disapproved by administrators. These responses can be a short textual answer or an uploaded picture. If a submission is approved, the player receives the points for their submission. Otherwise, the system

notifies the player that their submission was not approved, along with a comment (written by an administrator) explaining why it was rejected. The player can change and resubmit their response and still earn the full point value for that task.

Most activities have fixed reward points if completed. Creative activities are a special kind of activity that are worth a variable number of points, which depends on the effort made by the player and the quality of the outcome submitted. It is judged by the administrators. These activities enable players to exercise their artistic talents.

*Commitments* are pledges that the player will carry out a specific action for a specific amount of time (typically 5 days). Examples include: reducing shower time, taking the stairs, and turning off the lights when leaving a room. Unlike activities, commitments are not easily verifiable, and so they are usually designed with fewer points than activities. Furthermore, a player can only enter into five commitments at any given time. After the commitment period is up, the player can declare that they completed the commitment and immediately earn the associated points. They can then enter into another commitment, including the one they just completed.

*Events* are actions tied to real world meetings. To help organizers gauge interest in events, players can earn points by signing up in advance. Players that do this (and then actually attend the event) earn a signup bonus (typically 2 points). Players can also set up a reminder that is sent to their email and/or their mobile phone before the meeting takes place. At the event, a challenge administrator provides players with “attendance codes” printed on slips of paper that can be later entered in the system by the player to get their points. (The paper slips provide a form of verification that the player physically attended the event.) Attendance codes are generated by Makahiki and can only be used once. To discourage players from signing up and not attending, a penalty (typically 2 points) is assessed to players who do not submit an attendance code. If the player submits an attendance code for the event after receiving this penalty, the penalty is reversed.

To make your SGG more interesting to players, and more pedagogically sophisticated, Makahiki supports the definition of “*Path*” through the educational content or actions. In most cases, when a new player sees the SGG for the first time, there should only be a few actions available to them - possibly only one. All of the rest should be locked. Makahiki provide a set of predicates that can

be used to define the “path.” The predicates determines if an action or level is locked or unlocked for a player, which in term depends on the outcome of another action or multiple other actions. Predicates include: completed a certain number of actions within a category, completed all actions within a category, completed a certain action, and unlocking of an action or level after a certain date.

To incentivize players to work together during a challenge, Makahiki also supports “*Social Bonus*”. The social bonus is an optional attribute of any Smart Grid Game action which awards extra points if the player has done the action with someone else. Examples of actions which commonly include a social bonus are: attending an event, recording a song related to energy, or measuring a shower water flow rate. When a player submits a response for an action that supports the social bonus, the player can provide the email address of another player who jointly completed the action. Once the other player also completes the task, the social bonus is awarded. Social bonuses are not bi-directional; if the second player does not provide the first player’s email address, only the first player will get the social bonus.

Designing the Smart Grid Game is not easy; in fact, it is the most complicated task for a game designer when designing a serious game using Makahiki. Makahiki provides a tool called Smart Grid Game Designer to ease this task. The designer page is accessible only by administrators.

**Figure 3.33** shows the basic components of the designer widget interface:

The Designer Widget has three columns, Library Actions, Designer Grid, and Palette.

The Library Actions Column holds a library of Activities, Commitments, and Events. The Library is a reusable set of actions for a sustainability serious game. These actions are divided into three tabs, Activities, Commitments, Events. They are generic actions without any dates or locations associated with them. You can drag these library actions into the Designer Grid. If the Action is an Event you will be asked to provide the event date and location. After the action is dragged into the grid, you can override any of the attributes of the action to be tailored to the need of a specific instance of the serious game challenge.

The grid that the Smart Grid Game Designer creates is called “draft”. Designers can adjust the Designer Grid, adding or removing actions, columns and levels and players will not see the changes until they are published. Multiple drafts can be created by saving the current grid to a different

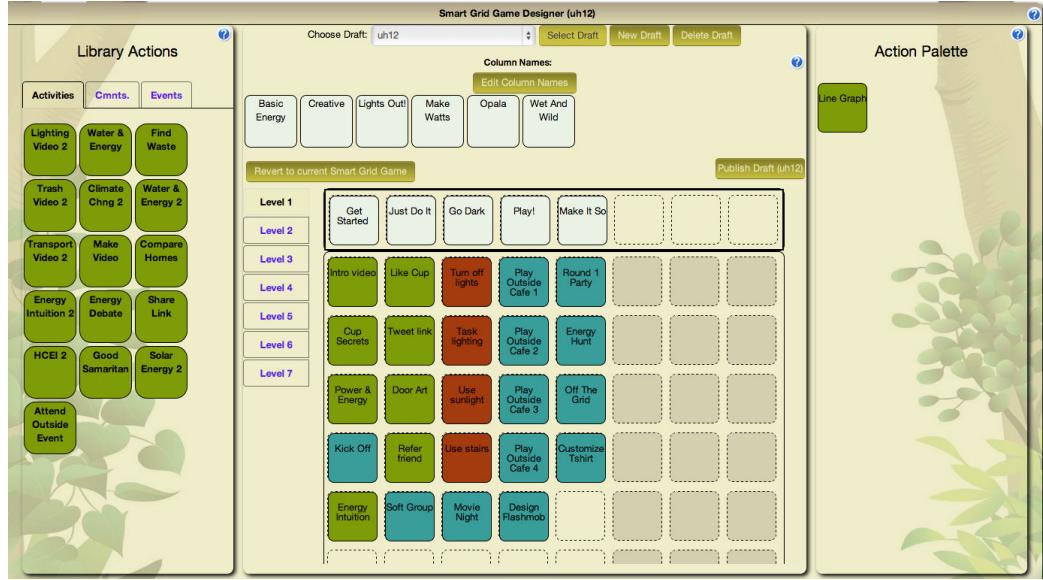


Figure 3.33: Smart Grid Game Designer widget

name. Multiple grids allow human game designers to explore different layouts and paths through the Smart Grid Game. Once the designer is satisfied with the draft grid, he can click the “publish draft” button to update the live Smart Grid Game that players will see.

The Smart Grid Game Designer also includes a Grid Consistency Checker (GCC) Tool. Because of the complexity of the grid, especially the interdependency (specified by the predicates) of the grid actions, a draft grid often contains errors and inconsistencies. The GCC will be automatically run when the “publish the draft” button is pressed and display all the potential errors in the grid. Users should modify the draft to fix any errors found by GCC in order to successfully publish to the live smart grid game.

**Figure 3.34** shows a list of items that checked by the GCC tool.

**Figure 3.35** shows the action dependency tree that is inspected by the GCC tool.



Figure 3.34: Grid Consistency Checker Widget



Figure 3.35: Action Dependency Tree

### 3.3.2.4 Top Score Game

The Top Score Game enables you to design a challenge in which prizes are awarded to individuals and teams who earn the most points during the challenge (and/or each round in a challenge). It also enables you to award prizes to the teams that conserved the most energy (or some other resource such as water) during the challenge (and/or each round in the challenge).

The Top Score Game includes both a Leaderboard widget (Figure 3.36(a)) that shows the top players/teams in contention for each round, and a Prizes widget (Figure 3.36(b)) that shows the prizes for each round and the current teams/individuals in line to win them.

Top Score Game (Team, Overall)		
Rank	Lounge	Points
1	Lehua-D	26849
2	Lokelani-C	21381
3	Mokihana-C	20904
4	Lokelani-A	18821
5	Ilima-B	15780
6	Mokihana-B	14536
7	Lehua-C	14024
8	Lehua-E	11270
9	Lokelani-E	10523
10	Mokihana-D	9473

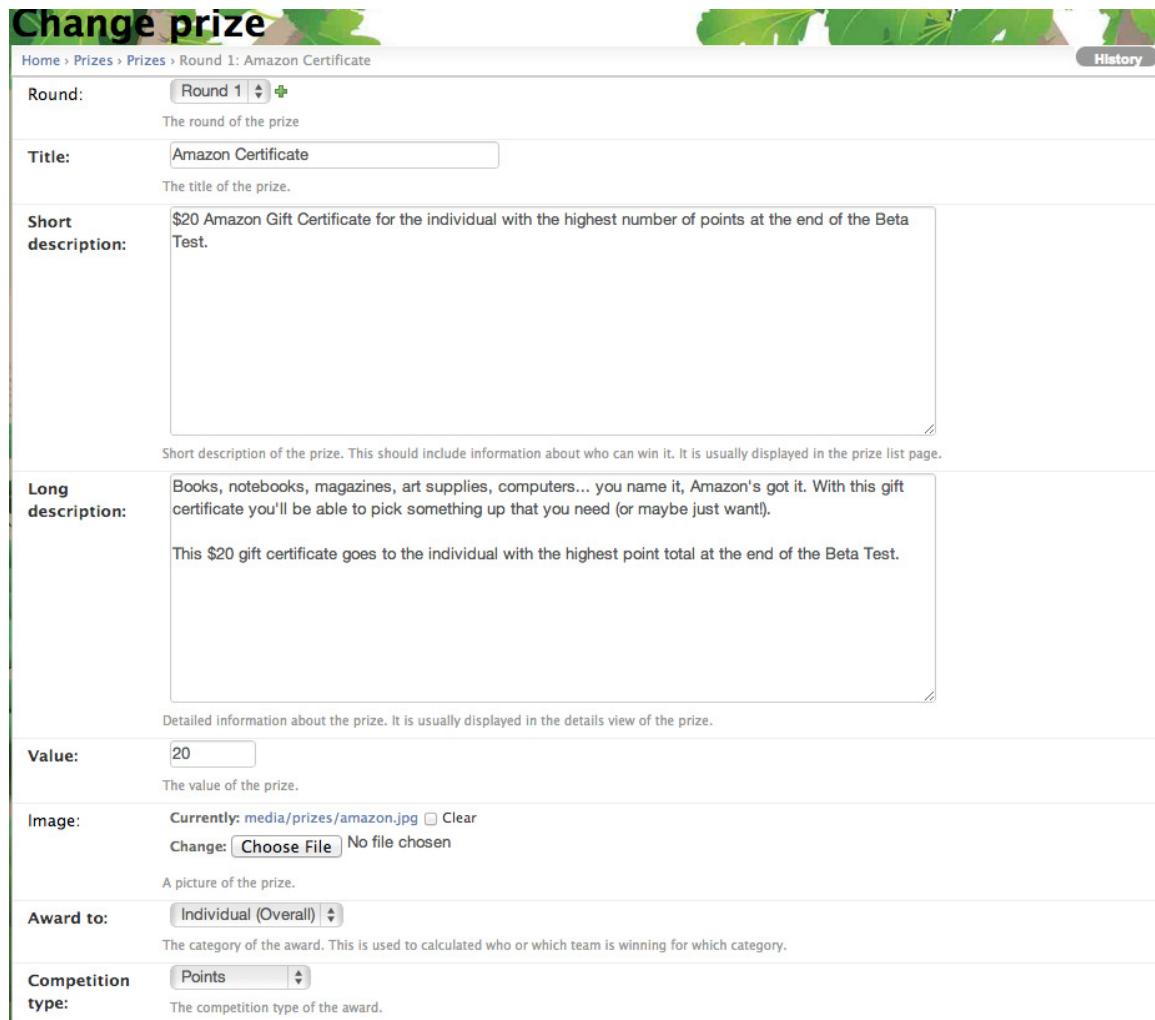
(a) TopScore Game Scoreboard



(b) TopScore Game Prizes

Figure 3.36: Top Score Game

Designers can configure the prizes for a particular challenge by using TopScore Game configuration on the “Settings” page. [Figure 3.37](#) shows the interface to specify a prize for the challenge.



The screenshot shows the 'Change prize' interface in a web-based game configuration tool. The top navigation bar includes 'Home', 'Prizes', 'Prizes', 'Round 1: Amazon Certificate', and a 'History' button. The main form is titled 'Change prize' and contains the following fields:

- Round:** Round 1 (dropdown menu with a plus sign icon)
- Title:** Amazon Certificate
- Short description:** \$20 Amazon Gift Certificate for the individual with the highest number of points at the end of the Beta Test.
- Long description:** Books, notebooks, magazines, art supplies, computers... you name it, Amazon's got it. With this gift certificate you'll be able to pick something up that you need (or maybe just want!).  
This \$20 gift certificate goes to the individual with the highest point total at the end of the Beta Test.
- Value:** 20
- Image:** Currently: media/prizes/amazon.jpg (checkbox labeled 'Clear')  
Change: Choose File (button) No file chosen
- Award to:** Individual (Overall) (dropdown menu)
- Competition type:** Points (dropdown menu)

Detailed information about the prize is displayed below the long description, stating: "Detailed information about the prize. It is usually displayed in the details view of the prize."

Figure 3.37: Specifying a TopScore Game Prize

### 3.3.2.5 Raffle Game

The Raffle Game was designed to solve the problem of incentivizing players who cannot hope to be a top competitor in the Challenge. When several hundred players are competing, only a handful have a realistic chance to be the top scorers for a round. Once a player knows they cannot beat all of the other players, there can be an urge to simply give up.

The Raffle Game is designed to enable all players to have a chance to win a wide variety of prizes, where their odds of winning increase based upon the number points they have earned in the game; thus incentivize participation from all individuals, even those who are not in the running for a top prize. [Figure 3.38](#) shows an example of the Raffle Game.

Round 2 Raffle Game					
Your total raffle tickets: 5 Allocated right now: 2 Available: 3					
Prize	Value	Your tickets	Total tickets	Current odds	Change ticket allocation
Recycled bike	\$200.00	1	2	50.0%	+1 -1
UH t-shirt (1)	\$28.00	1	1	100.0%	+1 -1
Outback card	\$25.00	0	0	0.0%	+1 -1
Smart strip (2)	\$25.00	0	0	0.0%	+1 -1
Smart strip (1)	\$25.00	0	0	0.0%	+1 -1
Down to Earth card	\$25.00	0	0	0.0%	+1 -1

Figure 3.38: Raffle Game widget

Each round of the competition has its own set of raffle prizes and any unused raffle tickets carry over to the next round. Raffle tickets are independent from a player's score, and allocating a raffle ticket does not affect their rank. The system provides random selection of the winner of each raffle item at the end of a round.

The Raffle Game works in the following way. For every 25 points (by default) that a player earns, they receive one virtual raffle ticket. Players can dynamically allocate their tickets to any raffle prizes they are interested in at any time, up to the end of the raffle. Each round of the competition

has its own set of raffle prizes and any unused raffle tickets carry over to the next round. Raffle tickets are independent from a player's score; allocating a raffle ticket does not affect their rank.

As the screen image above shows, each player can see in real-time how many Raffle Game Tickets they have earned, which prizes they have allocated them to, and the resulting percentage chance they have of winning based upon the tickets allocated by others to that same prize. Of course, these odds can change on a moment-to-moment basis as players allocate and deallocate tickets.

The Raffle Game, in addition to providing an incentive for the non-top players to earn points, also creates an incentive for players to come back to the site on a regular basis to see the updated odds associated with their choices.

Designers can configure the raffle prizes for a particular challenge by using the Raffle Game Configuration link on the “Settings” page. [Figure 3.39](#) shows the interface to specify a prize for the challenge.

The screenshot shows a web-based configuration interface for a raffle prize. At the top, there is a header bar with the title 'Change raffle prize' and a 'History' button. Below the header, the main form fields are displayed:

- Round:** A dropdown menu showing 'Round 1'.
- Title:** An input field containing 'Energy Meter'.
- Value:** An input field containing '30'.
- Description:** A text area with the following content:

Want to keep learning about your energy consumption even after the Kukui Cup is over? Bid on your very own Belkin Conserve Insight Energy Monitor!
- Image:** A section showing the current image file as 'media/prizes/belkin-energy-monitor.jpg' with a 'Clear' link and a 'Choose File' button.
- Winner:** A dropdown menu currently set to '-----'.

At the bottom of the form, there is a 'Raffle tickets' section showing a single entry:

User	Created at
admin	Jan. 29, 2013, 9:38 a.m.

Below the table are three buttons: 'Delete', 'Save and add another', 'Save and continue editing', and a large blue 'Save' button.

Figure 3.39: Specifying a Raffle Game Prize

The “Winner” of the raffle prize is randomly picked by the system at the end of the round with odds in proportion to each player’s relative allocated number of raffle tickets. At the end of the page,

you can also see a list of users that allocated raffle tickets for this raffle prize.

### 3.3.2.6 Participation Game

One of the design constraints of a sustainability serious game such as the Kukui Cup challenge is that, the players associated with each team in a challenge must be specified in advance of a challenge. Thus, as the challenge runs, it is possible to know exactly what percentage of each team's players are actively playing the game (in the sense that they have logged in at least once).

The Participation Game is designed to incentivize active players on a team to recruit other members of their team to login and try the game. It does this by providing extra points to all active players on a team when the percentage participation by that team reaches certain thresholds (currently 50, 75, and 100).

The current percentage participation by a player's team is shown in a scoreboard, as shown in the [Figure 3.40](#). Players will receive an in-game notification whenever they reach a threshold participation where points are awarded.

Overall Scoreboard		
Rank	Team	Participation
1	Ilima: Lounge A	74%
2	Mokihana: Lounge E	53%
3	Lokelani: Lounge A	51%
4	Lehua: Lounge D	44%
5	Lehua: Lounge E	43%
6	Mokihana: Lounge D	37%
7	Mokihana: Lounge A	35%
8	Mokihana: Lounge B	22%
9	Lehua: Lounge B	21%
10	Lokelani: Lounge B	16%

Figure 3.40: Participation Game Scoreboard

To configure the participation game, click on the “Participation Settings” link in the Participation Game Admin widget on the “settings” admin page. [Figure 3.40](#) shows the settings of the participation game. You can change the points to award for each participation percentage. Currently, the percentages (50, 75, and 100) are hardwired into the system.

**Change participation setting**

Home > Participation > Participation settings > ParticipationSetting object

Points 50 percent:	5	The point amount for 50 percent participation.
Points 75 percent:	5	The point amount for 75 percent participation.
Points 100 percent:	10	The point amount for 100 percent participation.

Save and continue editing | Save

Figure 3.41: Specifying a participation Game Settings

### 3.3.2.7 Quest Game Mechanics

One fundamental challenge faced by any game is: how do players learn how to play it? This is generally known as the on-boarding problem. Makahiki provides a configurable “Quest Engine”, that enables the definition of quests and the dependencies among them. That enables site developers to create a kind of structured, proactive user guide for the system. Instead of stumbling on a random playable feature of the game, players learn about the capabilities of the site by performing discrete sequences of actions, called “Quests.”

Quests are made available to the player in a collapsible/expandable window right below the navigation bar. The set of Quests shown to a player can depend upon their game state, so that “simple” Quests can be presented initially and more “complicated” Quests presented as the player gains in expertise. Quests generally guide the player through the various workflows of the Challenge, such as completing a task, signing up for an event, or allocating a raffle ticket.

The system shows a maximum of three Quests at a time. Figure 3.42 shows an example of an expanded quest and its content. Once the player accepts the quest, and follows the instructions to complete the quest, the notification dialog box will appear to the player indicating the accepted quest had been completed. A new potentially harder quest will appear in the quest window replacing the completed quest.

Quests are created by the administrator prior to the Challenge. Administrators have the option of specifying a set of predicates to determine when the player could be shown a Quest and when the player has completed the Quest, and it should no longer be shown.

Designers can configure the quests for a particular challenge by using Quest Game Mechanics Configuration link on the “Settings” page. Figure 3.43 shows the interface to specify a quest for the



Figure 3.42: Quest Game Mechanics

challenge.

### Change quest

Home > Quests > Learn Secrets of Kukui Cup Masters

Name:	<input type="text" value="Learn Secrets of Kukui Cup Masters"/>
The name of the quest.	
Quest slug:	<input type="text" value="learn-secrets"/>
Automatically generated if left blank.	
Description:	<p>Do you want some hints about how to win the Kukui Cup? We've prepared a special video for you, called "Secrets of the Kukui Cup Masters". Watch it to learn strategies for playing the Kukui Cup.</p> <p>**To complete this quest:**</p> <ol style="list-style-type: none"> <li>1. If you aren't already on the Get Nutz page, click on the Get Nutz icon in the navigation bar at the top of every page</li> <li>2. Click on the second square in the first column of the Smart Grid Game</li> <li>3. Watch the video</li> <li>4. Answer a question about it correctly</li> </ol> <p>Description of the quest. It should outline the steps to completing this quest. Uses Markdown formatting.</p>
Priority:	<input type="text" value="1"/>
Quest with lower values (higher priority) will be listed first.	
Unlock conditions:	completed_action('intro-video') and not completed_action('cup-secrets')
Conditions a user needs to meet in order to have this quest be available (appeared in the Quest widget). Uses Makahiki Predicates.	
Completion conditions:	completed_action('cup-secrets')
Conditions a user needs to meet in order to complete the quest. Uses Makahiki Predicates.	

Figure 3.43: Specifying a Quest Game Mechanics

### 3.3.2.8 Badge Game Mechanics

Makahiki provides badge game mechanics to motivate and engage players. Badge mechanics is implemented in a customizable way so that game designers can create as many badges as they like. Each badge can be triggered by the achievement of a certain award condition, which is defined by the flexible predicate system in Makahiki.

Badges are a common game mechanic, in which players receive recognition for various accomplishments. Makahiki allows the challenge designer to specify the set of badges available in a challenge, and to define new ones. The challenge designer has the option of making badges worth points. Finally, the designer can use the Makahiki predicate system to award a badge automatically (for example, when a player has completed a Level in the Smart Grid Game), or manually award the badge through administrator action (for example, when a player reports a significant bug in the system).

In many systems, each badge has a custom design, but in Makahiki, we decided that the overhead of providing a custom graphic for each badge outweighed the benefits. Providing a custom graphic also would create complexity with another feature of Makahiki: the ability to create “themes” with different colors. To be consistent with the themes in Makahiki, badges have a common look and feel consisting of a circle and a multi-character ID. Its actual colors are specified by the theme, and can thus vary from theme to theme.

[Figure 3.44](#) shows an example of the badges available in the Makahiki system:

Designers can configure the badges for a particular challenge by using Badge Game Mechanics Configuration link on the “Settings” page. [Figure 3.45](#) shows the interface to specify a badge for the challenge.

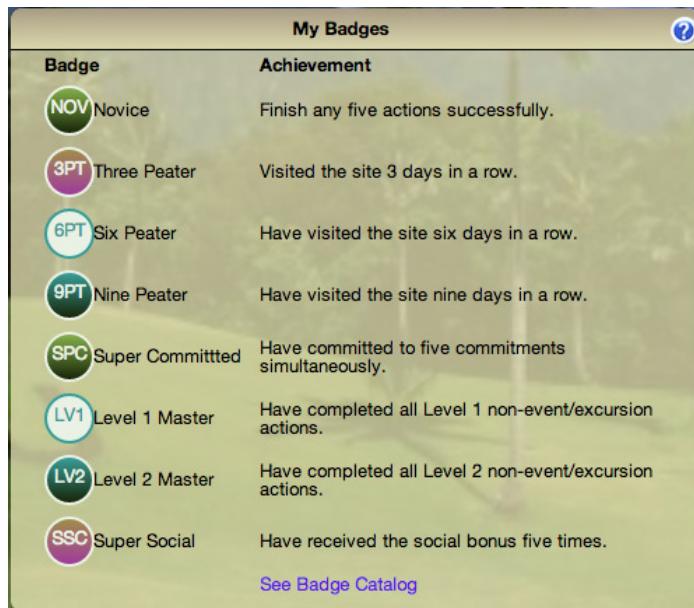


Figure 3.44: Badge widget

**Change badge**

Home > Badges > Badges > Three Peater

History

Name:	Three Peater	The name of the badge
Slug:	three-peater	A unique identifier of the badge. Automatically generated if left blank.
Label:	3PT	The label of the badge
Description:	Visited the site 3 days in a row.	The description of the badge
Hint:	Keep coming back!	The Hint of the badge
Points:	10	Points awarded for getting badge.
Priority:	20	sorting order in the badge list, lower value (higher priority) appears first.
Award condition:	daily_visit_count(3)	If the condition is True, the badge will be awarded. Uses Makahiki Predicates.
Award trigger:	daily	The trigger of evaluating the award condition.
Theme:	theme 2	The theme for the badge.

\* Delete      Save and add another      Save and continue editing      **Save**

Figure 3.45: Specifying a Badge Game Mechanics

### 3.3.2.9 Referral Bonuses Game Mechanics

Similar to the social bonus described in the Smart Grid Game above, the Referral Bonus is the game mechanics that help encourage participation by providing additional points to players who participate in activities with other players, and facilitate the entry of new players into an energy challenge.

Players are led through a setup process when logging into Makahiki for the first time. One of the steps in this process is the referral bonus. If a player was referred by another player in the system, they can use this step to input their email address. Once the new player earns a certain number of points in the competition, both players are awarded a referral bonus of a configurable number of points. Typically, going through the setup process gives you 25 points, so setting a point threshold of 30 points encourages the new player to at least complete one additional action in order to get the referral bonus.

When enabled, the referral bonus is implemented as a step in the first login process, as shown in [Figure 3.46](#).

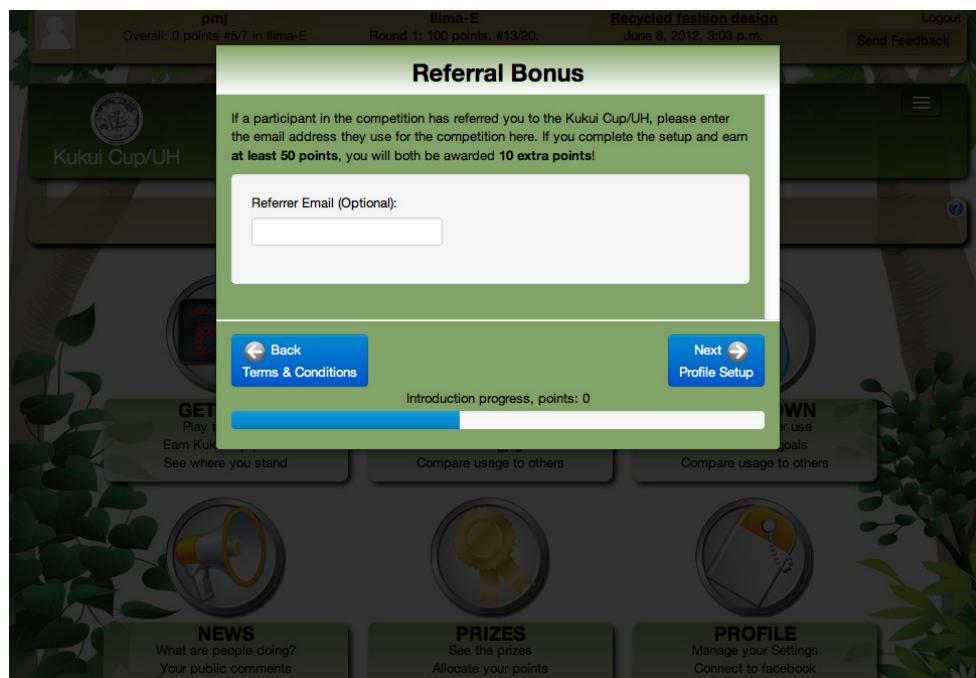


Figure 3.46: Referral Bonus Game Mechanics Widget

If the new player was referred to the challenge by another player, they can use this step to input their email address. Once the new player earns 30 points in the competition, both players are awarded a referral bonus of (typically) 10 points. Typically, going through the setup process gives you 25 points, so a threshold of 30 points means the new player has to complete at least one additional task in order to get the referral bonus.

You can disable the referral game mechanics by clicking on the “Referral Game Mechanics” link. If referral game mechanics is disabled, then this window is omitted from the first login wizard and players will not be able to get points by referring other players.

The referral bonus also has a “dynamic bonus” capability. If enabled, then you can vary the amount of points awarded depending upon the participation level of the team associated with the new player. This incentivizes players to not just recruit new players for their own team, but to also recruit players for other teams who might not have much participation.

If the referral game mechanics is enabled, which is true by default, you will see the “Referral settings” link in the Game Admin widget. After clicking on the “Referral Settings” link, you will see the overview of the referral settings; clicking on any of the links, will bring you to a page ([Figure 3.47](#)) to change the settings:

The screenshot shows a web-based configuration interface titled "Change referral setting". At the top, there's a breadcrumb navigation: Home > Score\_mgr > Referral settings > ReferralSetting object. On the right side of the header is a "History" button. Below the header, there are three input fields for different referral types:

- Normal referral points:** A text input field containing "10" with a note below it: "The point amount for normal referral bonus."
- Super referral points:** A text input field containing "20" with a note below it: "The point amount for supper referral bonus, when the referral is from a team of participation rate from 20% to 40%".
- Mega referral points:** A text input field containing "30" with a note below it: "The point amount for mega referral bonus, when the referrals is from a team of participation rate les than 20%".

Below these fields is a checkbox labeled "Start dynamic bonus" with the following description: "Start rewarding the dynamic referral bonus. set it to true if you want to reward referral bonus depends on referral's team participation." To the right of the checkbox are two buttons: "Save and continue editing" and a larger blue "Save" button.

Figure 3.47: Changing the Referral Bonus Game Mechanics Settings

By default, only the “Normal” referral points value is used. If you check the “Start dynamic bonus” setting, then the “Super” and “Mega” values are enabled depending upon the team participation rate of the new player.

### 3.3.2.10 Configuring the games in a page

Once the games and mechanics are selected, a challenge designer can decide which games to include in a page and how the game widgets appear. He can use the “Page info” link in the “Challenge Design” admin interface to configure the game widgets and the layout of a page. [Figure 3.48](#) shows the interface for configuring a page.

A designer can specify the location of the widgets in the page. Currently Makahiki only supports the two-column layout in a page. The “Location” setting specifies whether the widget will appear in the left column or right column. The “Priority” setting specifies the vertical order of the widget in the column. Lower priority numbers will appear closer to the top of the column. The “Enable” field specifies if the widget appears or not.

In addition to the configuration for the widgets, a designer can also configure the name of the page, whether the page should appear or not, and the location order of the page in the navigation bar.

The screenshot shows the 'Change page info' interface. At the top, there's a title 'Change page info' and a breadcrumb trail 'Home > Challenge\_mgr > Page infos > learn'. A 'History' button is in the top right. The main area has several input fields:

- Label:** Get Nutz (disabled)
- Title:** Get Nutz (disabled)
- Introduction:**
  - \* Play the Game
  - \* Earn Kukuicup points
  - \* See where you stand
- Unlock condition:** True (disabled)
- Priority:** 2

Below these is a 'Page widgets' section with a table:

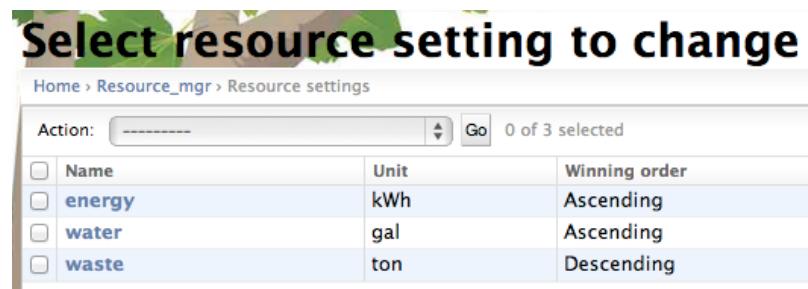
Widget	Location	Priority	Enabled	Delete?
smartgrid	Left	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
upcoming_events	Left	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
scoreboard	Right	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
participation	Right	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>

At the bottom are buttons: 'Delete' (disabled), 'Save and add another', 'Save and continue editing', and a large blue 'Save' button.

Figure 3.48: Configuring a Page

### 3.3.3 Configurable resource

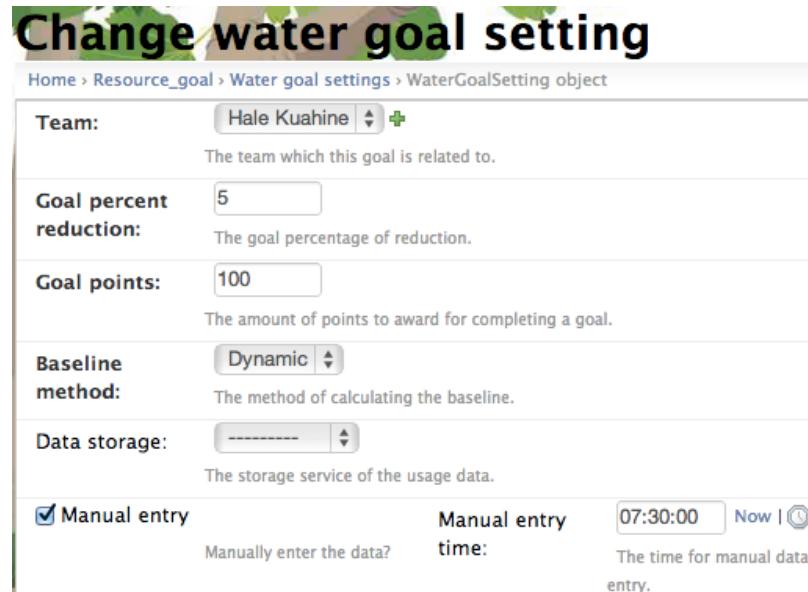
In Makahiki, different resources can be tracked and configured. The admin interface is built in to support the configuration of different resources. Makahiki supports three kinds of resources, which have different attributes: energy, water and waste. Some resource data can be obtained automatically from smart meters, while some resource data has to be input manually. In the case of manually data entry, the time of manual entry can be configured as well. [Figure 3.49](#) shows the Makahiki admin interface to configure the resources.



The screenshot shows a table titled "Select resource setting to change". The table lists four resource types: energy, water, and waste. Each row includes a checkbox, the name, unit, and winning order. The "energy" row is selected, indicated by a blue background.

<input type="checkbox"/>	Name	Unit	Winning order
<input checked="" type="checkbox"/>	energy	kWh	Ascending
<input type="checkbox"/>	water	gal	Ascending
<input type="checkbox"/>	waste	ton	Descending

(a) Supported resource types



The screenshot shows a form titled "Change water goal setting". It includes fields for Team (Hale Kuahine), Goal percent reduction (5), Goal points (100), Baseline method (Dynamic), Data storage, and Manual entry settings. The "Manual entry" checkbox is checked, and the manual entry time is set to 07:30:00.

Team:	Hale Kuahine
Goal percent reduction:	5
Goal points:	100
Baseline method:	Dynamic
Data storage:	-----
<input checked="" type="checkbox"/> Manual entry Manually enter the data?	
Manual entry time:	07:30:00 Now

(b) Manual resource

Figure 3.49: Configurable resource

Makahiki supports both automated and manual data collection. With respect to automated energy collection, Makahiki queries a WattDepot server once an hour to get an update on each team's

consumption during the previous hour, and then updates the stoplight visualization. At midnight, Makahiki determines whether the conservation goal was achieved by the team and updates the calendar-based view with the results for that day.

However, not all challenge player communities have meters that are internet-accessible and thus allow this kind of real-time, automated update. Instead, they might have a traditional, analog meter.

The Energy Goal Game can be configured to support manual data collection. To accomplish this, the challenge designers must first tell the system the time each day at which they will read the meters manually. (To make the energy goal game workable, the challenge designers must commit to reading the energy meters for each team at approximately the same time each day so Makahiki can assume the data represents equal, 24 hour intervals. Team meters can be read at different times, but the time must be consistent for each team.)

Then, each day during the challenge, the challenge designers read the meters, then login to the system and update Makahiki with the latest readings. From this, Makahiki can determine which teams made their energy goal for the previous day.

From a user interface perspective, the basic difference is that the stoplight visualization is not available. Instead, the primary interface to the Energy Goal Game is the calendar-based visualization, which shows the results for each day.

Given support for both automated and manual energy data collection, Makahiki can also support Water Goal Games, Food Goal Games, Waste Goal Games, or any other “resource” for which teams are responsible. Currently, Makahiki provides built-in support for two resource goal games: energy and water. Each of those games, when enabled, results in a page devoted to that resource in the web application. The default configuration enables support for the Water Goal Game and the Water page.

Extending Makahiki to support an additional resource goal game is straightforward, but requires developer-level capabilities.

### 3.3.4 Real-time Analytics

Makahiki is designed to support sustainability challenges involving hundreds or thousands of users lasting weeks or months. In these circumstances, effective use of the technology requires the ability to understand the state of the game, such as: Who is using it? What are they doing? What is the player response to activities, commitments, excursions, and events? Such state information is important for planning purposes, such as assessing the transportation needs for an upcoming excursion by seeing how many players signed up. It can also be used for making in-game changes to game design, such as changing the point values associated with activities to encourage or discourage participation. It can also help identify breakdowns in game play, such as significant numbers of unallocated raffle tickets indicating that users do not understand the nature of that game mechanic.

To address these needs and others, Makahiki includes a variety of widgets that work together to provide high level overview of game play state to the administrators of a challenge. [Figure 3.50](#) shows an example of two game analytic widgets.

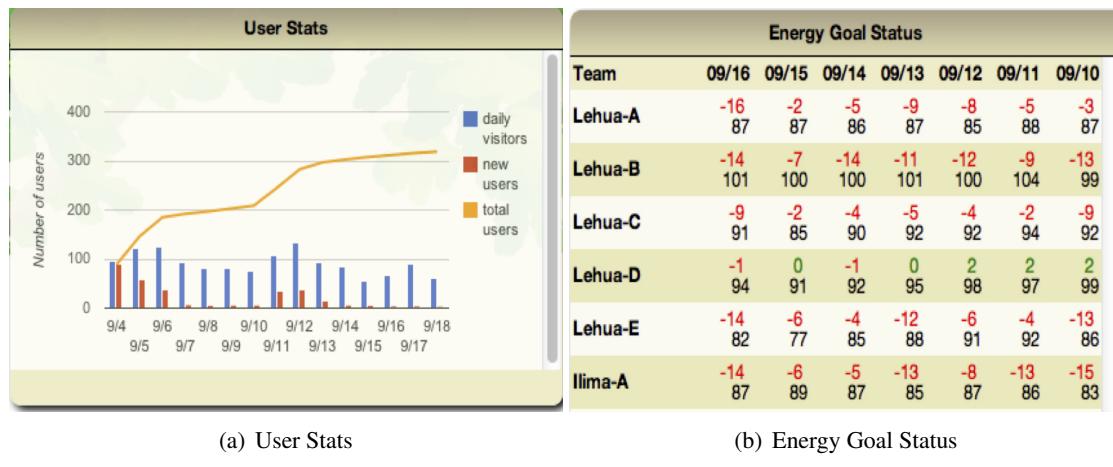


Figure 3.50: Game analytic widgets: User Stats and Energy Goal Status

The left widget, User Stats, shows trends in the total number of players, the total number of new users, and the total number of players visiting the site each day. The right widget provides information on the ability of teams to achieve their daily energy goal each day and over time.

[Table 3.3](#) lists all the analytics widgets that are currently included on the “Status” page. More analytics widgets can be developed using the widget development API and added to the page.

<b>Analytics Widget</b>	<b>Display Content</b>
Player points	Rankings of all the players' points for each round
Team points	Rankings of all teams' total points for each round
Team participation	Rankings of all teams' participation rate for each round
Team energy	Rankings of all teams' energy usage and goal meeting status for each round
Team water	Rankings of all teams' water usage and goal meeting status for each round
Raffle	Number of tickets for each raffle for each round
Current status	The status of the submission approval queue; number of unallocated raffle tickets
Event RSVPs	Number of RSVPs for each event
Popular event	Number of signups and completions for each event ordered by signups
Popular activity	Number of submissions and approved for each activity ordered by submissions
Popular commitment	Number of signups and completions for each commitment ordered by signups
Popular quests	Number of completed quests ordered by completions
Feedback (rating)	Average ratings and total number of player feedbacks for each activity
Awarded badges	Number of players awarded for each badge
Referrals	Number of referrals for the players that made a referral
User stats	Graph showing the total number of players, number of new users, and number of players visiting the site for each day
Energy goal status	Status of all teams' meeting energy goal each day as well as the actual usage and the difference between the usage and goal
Water goal status	Status of all teams' meeting water goal each day as well as the actual usage and the difference between the usage and goal
WattDepot status	Number of seconds taken to retrieve the Wattdepot server data for each team.

Table 3.3: Analytics Widgets on “Status” Page

### 3.3.5 Responsive mobile support

We believe that mobile support is essential for this kind of sustainability challenge, especially for the new generation players. Makahiki implemented responsive web design technology to support multiple devices to enhance the player experience. [Figure 3.51](#) shows the responsive interface in Makahiki that supports both desktop view and mobile view with the same code base.

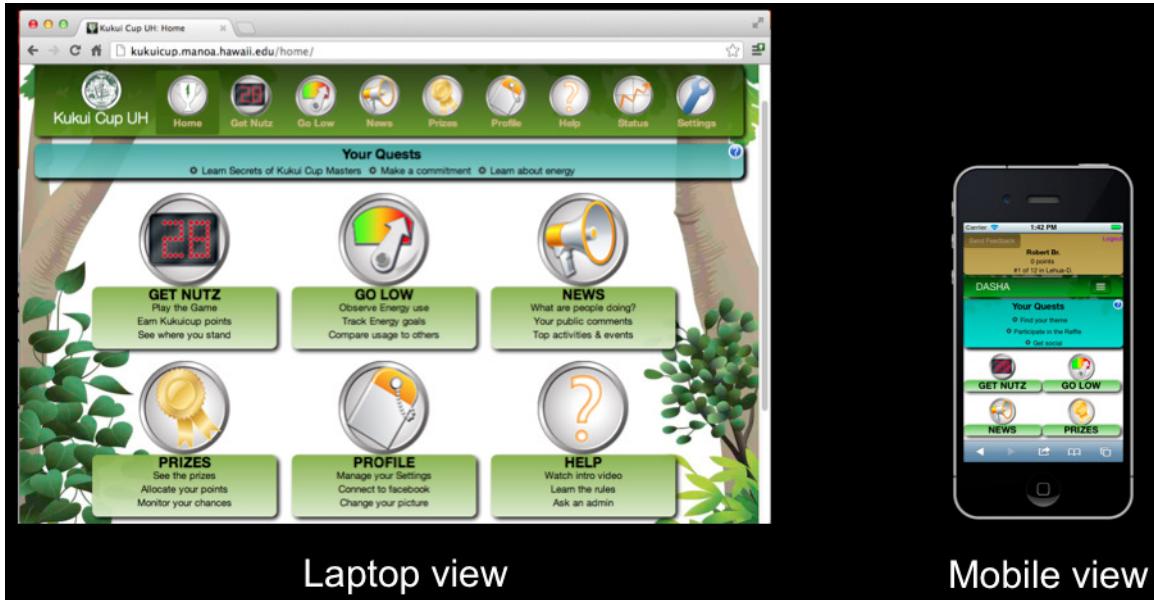


Figure 3.51: Responsive design supports both desktop and mobile

### 3.3.6 Cloud deployment support

Another feature in Makahiki is the ability to deploy to a Cloud platform. Cloud computing has the advantage of simplifying IT administration by eliminating the need to acquire the hardware, install software etc, thus lowering the cost of software deployment. Figure 3.52 shows a screen shot of the Dashboard showing the 2012 East West Center Kukui Cup challenge deployed in Heroku, a cloud platform provider. The monthly cost for the IT infrastructure in this instance is quite affordable.

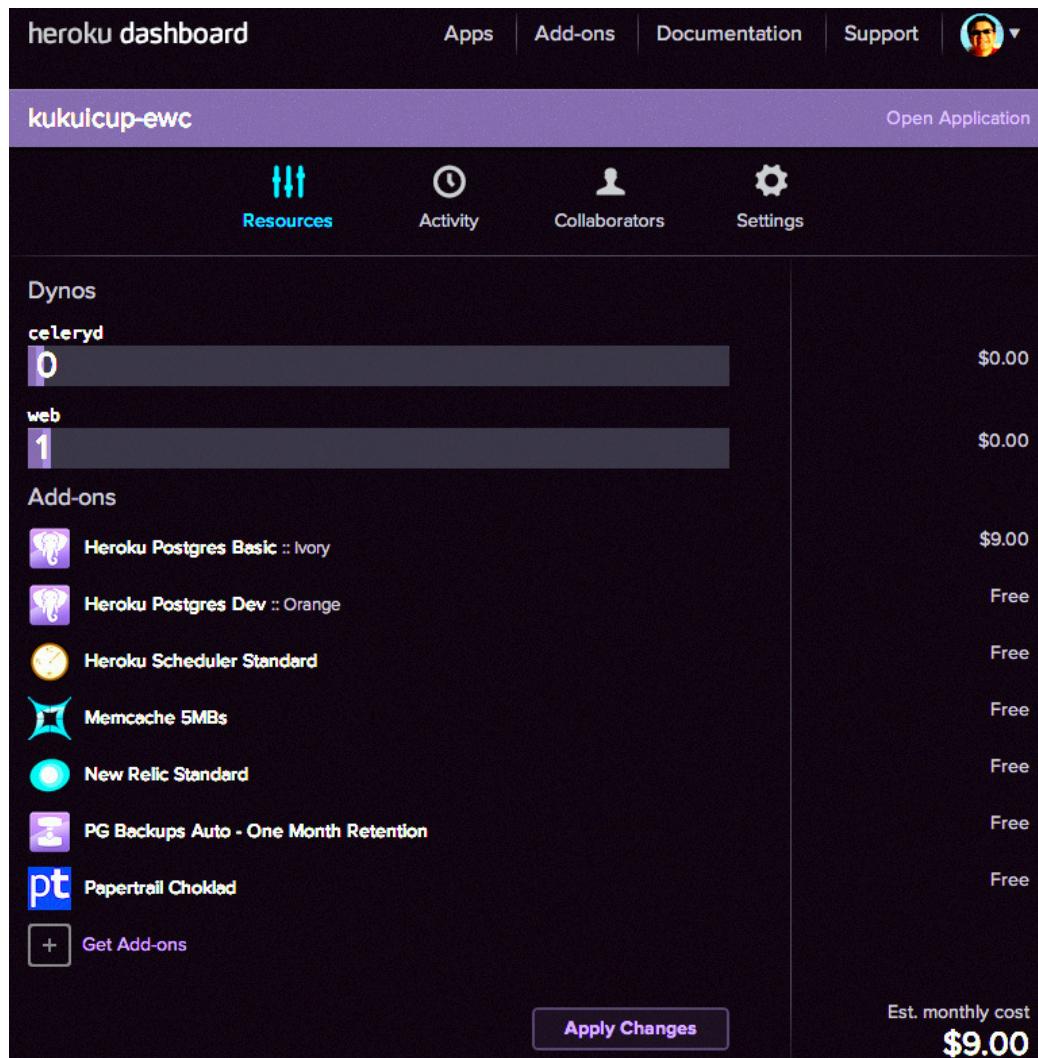


Figure 3.52: Heroku cloud deployment

Section 6.1.3 describes in details our experiences with several real world Makahiki instances that are deployed to the Heroku cloud infrastructure. In general, for a small organization, a monthly

cost of \$10 for hosting a Makahiki game instance is sufficient, such as in the East West Center case. For a large organization with about 1000 eligible players, the monthly hosting cost will be around \$100. See [Section 6.1.3](#) for more discussions on the cost of cloud hosting.

As discussed in [Section 6.1.3](#), besides the cost benefit, cloud deployment has the advantage of dynamically adjusting the computing resource for the application in both scaling up and scaling down scenarios.

### 3.3.7 Customization and Extension Development Support

Makahiki, as a framework, provides facilities to help developers to create customizations and extensions to fit different requirements from the organization. The following section describes the two customizations that can be implemented by developers through minimal programming.

#### 3.3.7.1 Theme Development

A “theme” in Makahiki consists of a specification of the background image (or color), as well as the background and font colors for various structural elements of the system. Players have the ability to select themes from the Profile page, making it unnecessary to develop a “perfect” theme for your challenge. [Figure 3.53](#) shows the widget that a player can use to change the theme they like the best for their personal website feel.

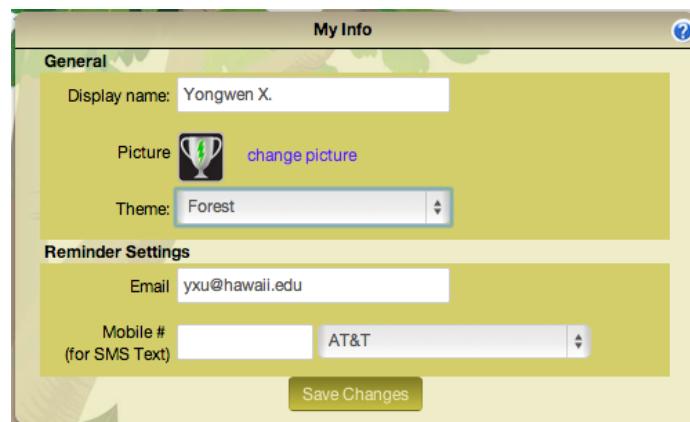


Figure 3.53: Change the Website Theme

Makahiki provides a set of pre-built themes for players to choose from. [Figure 3.54](#) lists all the

installed themes in Makahiki. These installed themes will show up in the “Theme” drop-down that can be selected by a player as shown in [Figure 3.53](#).

```
#####
# INSTALLED Themes. Please keep them in alphabetical order
#####
INSTALLED_THEMES = (
    'theme-bubbles',
    'theme-bumblebee',
    'theme-forest',
    'theme-google',
    'theme-hpu',
    'theme-revolusun',
    'theme-sonora',
    'theme-space',
    'theme-wave',
)
```

Figure 3.54: Makahiki Pre-Built Themes Defined in settings.py

Makahiki, as a framework, allows developers to create additional themes. Makahiki’s themes are implemented using LESS, which is the enhanced version of CSS that supports variables and other capabilities not available in standard CSS. There is no need to know about LESS in order to create Makahiki themes. Most developers should be able to create themes simply by copying and editing a pre-existing theme file and changing some of the LESS variables in the new file. All the theme files are located in the “makahiki/static/less” directory. Most of the times, a theme developer only needs to pick the color for the theme and specify which color he want for a specific UI elements in the LESS file. [Figure 3.55](#) shows a snippet of the theme-google.less file.

Once the theme file is created, it should be included in the “INSTALLED\_THEMES” section in the settings.py file as shown in the previous [Figure 3.54](#). To see the results of the new theme, Makahiki provides a special page called “theme-display”. The developer can use the browser to go to this page to test all of the theme-able elements in a single page. [Figure 3.56](#) illustrates the result of the new theme.

### 3.3.7.2 New Game Widget Development Support

As discussed previously, Makahiki includes a set of pre-built game and game mechanics widgets for game designers to select and configure for their serious game challenges. In some cases, game de-

```

// Google color palette
@google-white:           #FFFFFF;
@google-offwhite:        #E9EEF5;
@google-lightblue:        #8DAAEB;
@google-gold:             #E8AC13;
@google-lightlightblue:   #B8CAE0;
@google-darkblue:         #1249E0;
@google-red:              #D41C34;

// Page Background
@use-bkg-image: false;
// @page-bkg-image: "../images/forest-theme-background.jpg";

@page-bkg-color-start: @google-white;
@page-bkg-color-end: @google-white;
@page-font-color: @black;
@page-link-color: @google-darkblue;

// Widgets
@widget-title-bkg-color-start: @google-lightblue;
@widget-title-bkg-color-end: @google-lightblue;
@widget-title-font-color: @black;
@widget-title-transparency: 0%;
@widget-body-bkg-color: @google-white;
@widget-body-font-color: @black;
@widget-body-transparency: 0%;

// Smart grid game
@sgg-header-bkg-color: @google-offwhite;
@sgg-header-font-color: @black;
@sgg-entry-font-color: @black;
@sgg-activity-cell-bkg-color: @google-lightblue;
@sgg-commitment-cell-bkg-color: @google-red;
@sgg-event-cell-bkg-color: @google-gold;
@sgg-excursion-cell-bkg-color: @google-darkblue;
@sgg-filler-cell-bkg-color: lighten(@black, 30%);

```

Figure 3.55: Google Theme defined in theme-google.less

signers may wish to implement additional games or mechanics and integrate them into the Makahiki system. It is achievable through the game widget development support of the Makahiki framework. In order to develop new games, programming knowledge of Python and Django is required. This is unlike the theme development where there is no need to know LESS or any programming.

Developing a new widget for Makahiki includes the following four steps:

The first step is to create the new widget package structure. Makahiki provides a command line tool called “startwidget” integrated into the standard Django manage.py utility. It is the easiest way to create the basic file structure for a new widget. [Figure 3.57](#) illustrates the command to create a



Figure 3.56: The Result of theme-google.less

hello\_world widget and the base files and directory structure that are created by the tool.

```
% manage.py startwidget hello_world
```

```
hello_world/
    __init__.py
    views.py
    tests.py
    templates/
        index.html
```

Figure 3.57: Makahiki Tool for Creating a New Widget Structure

The command creates a directory named after the new widget in the directory “apps/widgets”, and four files underneath:

- `__init__.py`: this file indicates that the directory is a Python package.

- *views.py*: this file implements the widget logic and provide data to the UI.
- *index.html*: this file defines the UI that is displayed to players.
- *tests.py*: this file provides the skeleton of the unit testing for the new widget.

These files are the starting points for building a new widget.

The second step is to implement the logic of the widget and provide the data for display. It is done in the *views.py* by calling the Application Programming Interfaces (APIs) provided by Makahiki.

The Makahiki API provides a generic mechanism for supplying data to display in the widget UI. When the player loads a page, the Makahiki module “**apps.pages.views.index**” is called. This module determines the name of the page dynamically and creates a Python dictionary, called “**view-objects**”. From the page configuration, it determines which widgets are enabled for the given page. It then loops over each widget and calls their “**apps.widgets.widget\_name.views.supply**” function. So the entry point for the widget is the supply function in its *views.py* file.

The *startwidget* command provides the empty supply function in the *views.py* it created, as shown in [Figure 3.58](#).

```
"""Provides the view of the widget."""
def supply(request, page_name):
    """supply view-objects for widget rendering."""
    _ = request
    _ = page_name
    return {}
```

[Figure 3.58](#): Supply function template provided in the base *views.py*

A developer needs to fill in the supply function to provide the data. Makahiki also includes a library of manager modules that provides APIs to interact with the system. [Table 3.2](#) lists all the manager modules and their functionality.

For illustration purpose, the *hello\_world* widget will display three kinds of data obtained from the Makahiki system. One is the name of the player, which is stored in the *name* attribute of profile object. The second is the name of the team the player belongs to. It is stored in the *team* attribute of the profile object. The third data is the current point score of the player. It can be retrieved by

calling the function “player\_points” in the “score\_mgr” module. [Figure 3.59](#) shows the complete example of views.py for hello\_world widget.

```
"""Provide the view for the Hello-World widget."""
from apps.managers.score_mgr import score_mgr
def supply(request, page_name):
    """Supply view-objects contents, which are the player name, team and points.
    """
    _ = page_name
    profile = request.user.profile
    name = profile.name
    team = profile.team
    points = score_mgr.player_points(profile)
    return {
        "name": name,
        "team": team,
        "points": points,
    }
```

Figure 3.59: Example of a completed views.py

The third step in creating a new widget is to declare the UI component. It is done in index.html under the templates directory. [Figure 3.60](#) shows the base index.html created by the “startwidget” command.

```
<div class="content-box">
    <div class="content-box-title">
        Widget name
    </div>
    <div class="content-box-contents">
        Widget content
    </div>
</div>
```

Figure 3.60: Generated base index.html

Makahiki provides many different styles and CSS classes. Normally, widgets are contained in a content-box. A context-box is a rounded, shaded box with two parts, content-box-title, and content-box-contents. A developer can replace the “Widget name” in the title box and the “Widget content” in the content box to display the data provided from views.py. A complete example of the index.html file of the hello\_world widget is shown in [Figure 3.61](#). The data provided from the views.py are referenced by the convention of “**view\_objects.widget\_name.variable**”.

Finally, the fourth step for developing a new widget is to add the widget to a page. In order for

```

<div class="content-box">
    <div class="content-box-title">
        Hello World Widget
    </div>
    <div class="content-box-contents">
        Welcome <b>{{ view_objects.hello_world.name }}</b>, <br/>
        you're in team <b>{{ view_objects.hello_world.team }}</b> <br/>
        you have <b>{{ view_objects.hello_world.points }}</b> points .
    </div>
</div>

```

Figure 3.61: Example of a completed index.html

a new widget to be available to the system, you need to edit the Makahiki “settings.py” file and add the widget name to the INSTALLED\_WIDGET\_APPS variable. [Figure 3.62](#) shows a portion of the settings.py file after adding the new hello\_world widget.

```

#####
# INSTALLED Widgets
#####
INSTALLED_WIDGET_APPS = (
    'action_feedback',
    'ask_admin',
    'badge_scoreboard',
    'badges',
    'bonus_points',
    'hello_world',
    'home',
    ...
)

```

Figure 3.62: Adding the new widget to settings.py

This last step makes the newly developed widget available to the Makahiki system. Once it is available to the system, a game designer will be able to add the widget to an existing page using the admin interface.

The “Page Settings” link in the admin interface is used to configure the widgets in a page. [Figure 3.63](#) shows the hello\_world widget being added to the existing Profile page.

The hello\_world widget is added to the left side location with the priority set to “1”, which indicates that the new widget will be displayed first and before the “My Info” widget (priority 2) on the left side column on the page. The widget will also need to be enabled by checking the “enabled” checkbox. [Figure 3.64](#) shows the final result of the hello\_widget displayed in the Profile page.

Widget	Location	Priority	Enabled	Delete?
hello_world	Left	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
my_info	Left	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
badges	Right	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
my_commitments	Right	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
my_achievements	Right	3	<input checked="" type="checkbox"/>	<input type="checkbox"/>

[Add another Page Widget](#)

Figure 3.63: The Hello World widget added to the left column

Figure 3.64: The Hello World widget displayed in the Profile page

## 3.4 Summary

Makahiki is designed to provide an easy to use framework for different organizations to create custom sustainability related serious games.

Makahiki provides a libraries of games and game mechanics for organizations to choose and customize from. [Table 3.4](#) lists the Makahiki built-in games and game mechanics.

Name	Description	Type
Energy Game	awards points to players depending upon their ability to lower their energy consumption	game
Water Game	awards points to players depending upon their ability to lower their water consumption	game
Smart Grid Game	awards points to players for successfully completing activities, commitments, excursions, and events	game
Top Score Game	awards prizes to players and teams for earning the highest number of points during a round	game
Top Score Game	awards prizes to players and teams for earning the highest number of points during a round	game
Raffle Game	awards prizes to players if they have allocated their raffle tickets to a particular raffle prize	game
Participation Game	awards points to players if they can successfully get a certain percentage of their team members to participate	game
Quest	provides a way for players to learn about features of the challenge by guiding them through “Quests”	game mechanic
Badge	provides a way for players to earn badges for playing the game in a variety of ways	game mechanic
Referral	provides a way for players to earn points by getting other people to participate in the challenge	game mechanic

Table 3.4: Makahiki built-in games and mechanics

Using the provided different admin interfaces, system admins, game designers, and game developers can configure the Makahiki in various ways. In addition, players of Makahiki can also customize the look and feel (theme) of the game once they login. [Table 3.5](#) shows the customizable areas in Makahiki from the perspective of different roles. The game managers also use the admin interface to manage and monitoring the game. Currently Makahiki does not provide customization from a game management perspective. [Section 7.3.5.3](#) discusses a future Makahiki enhancement for this kind of customization.

User	Customizable area
Player	1. website look and feel (theme) 2. upload their own profile picture
System admin	1. methods of authentication and server configuration 2. email server configuration 3. energy data connectivity
Game designer	1. global game settings: branding, logo, default theme, landing page 2. competition settings: teams, rounds, durations 3. selection and configuration of individual games 4. manual or automatic resource data source 5. education “action” contents 6. levels, layout and progression of smart grid game
Game developer	1. location of a game widget in a page 2. adding new widgets

Table 3.5: Configurability in Makahiki

The extensive customization capability of Makahiki gives organizations great flexibility in tailoring a Makahiki serious game to suit their organizational needs.

## CHAPTER 4

# SGSEAM DESIGN

This chapter describes the design of the Serious Game Stakeholder Experience Assessment Method (SGSEAM) for assessing serious game frameworks. It starts with an overview of the SGSEAM process, followed by a discussion of the assessment methodology, and the detailed steps of the assessment method.

### 4.1 Overview

One of the benefits of using a serious game framework such as Makahiki, is that if correctly designed, it will provide useful and reusable “building blocks” with which to develop a variety of serious games. Yet how are we to know if a serious game framework has been “correctly designed”? As discussed in [Section 2.7](#), there are a few tools for general purpose assessment of software systems, but I have not found any prior work concerning the comprehensive approach to assess serious game frameworks. This is the motivation for the Serious Game Stakeholder Experience Assessment Method (SGSEAM). It is designed especially for assessing serious game frameworks in particular.

The Serious Game Stakeholder Experience Assessment Method (SGSEAM) describes a method for assessing serious game frameworks from the stakeholder experience perspective. We consider SGSEAM to be an assessment method instead of an evaluation method. The main purpose of an evaluation is to ”determine the quality of a program by formulating a judgement” [?]. An assessment, on the other hand, is nonjudgmental. SGSEAM does not try to judge a framework according to a standard, instead, it is used to identify the major strengths and shortcomings of a framework. The benefits of SGSEAM assessment are for the developers of serious game frameworks to learn and improve from the findings of the assessment.

The approach that SGSEAM uses is to assess the experiences of various important stakeholders when they interact with the serious game framework. In the full life cycle of a serious game framework there are a great variety of potential stakeholders, including:

- **Players:** those who participate in the game produced by the framework.

- **System admins:** those who install and maintain the technological game infrastructure.
- **Game designers:** those who design the content and game mechanics. They include content experts, instructional designers, etc.
- **Game managers:** those who manage the game during the period of game play.
- **Game developers:** those who use the game framework to customize, extend and enhance their games.
- **Researchers:** those who are conducting research using the game framework.
- **Spectators:** those who do not participate in the game play but are interested in the game and the results of game play.
- **Community partners:** those who partner with the game organizers to help run the game (such as coordinating real-world events as part of the game, providing support for data collection if the serious game requires data, etc.)
- **Funding organizations:** the organizations who provide funding for the game or game framework.

The scope of SGSEAM is to assess serious game frameworks as software infrastructure. While the overall success of a serious game depends on the individual success of all of these stakeholders, SGSEAM only assesses the experiences of the players, system admins, game designers, game managers, and game developers, because these are the stakeholders whose experiences are impacted by the software infrastructure.

**Figure 4.1** illustrates the three steps required to apply SGSEAM to a serious game framework.

1. Step one is to **Plan the assessment**, including identifying the stakeholders, determining assessment approaches, and creating the assessment schedule. The deliverable for this step is the *assessment plan* document.
2. Step two is to **Gather data** by carrying out the assessment, recording and obtaining related data. The deliverable for this step is the assessment *data repository*.

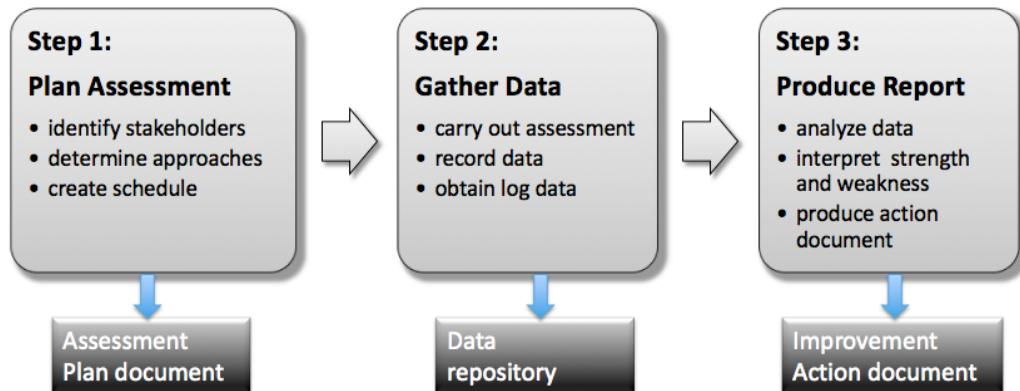


Figure 4.1: Applying SGSEAM to a serious game framework

3. Step three is to **Produce the assessment report** by analyzing the data and interpreting strengths and weaknesses. The deliverable for this step is the *improvement action* document.

The following sections describe the methodology used in SGSEAM, followed by a description of the steps required to apply SGSEAM to a serious game framework.

## 4.2 Methodology

SGSEAM is an assessment method instead of an evaluation method. The main purpose of an evaluation is to determine the quality of a program by formulating a judgment. An assessment, on the other hand, is nonjudgmental. SGSEAM does not try to judge a framework according to a standard, or to compare one framework against another. Instead, it is used to identify the major strengths and shortcomings of a framework to benefit the developers of the framework.

Creswell [?] categorizes research methods into three approaches: quantitative, qualitative, and mixed methods, according to what knowledge claims are being made and how knowledge is acquired. Quantitative methods reflect a post-positivist paradigm where hypotheses are specified *a priori* and tested by experimental design. Qualitative methods reflect a constructivist or participatory paradigm where knowledge is acquired by observation and open-ended design. SGSEAM employs the mixed methods approach which is based on pragmatic knowledge claims and the assumption that collecting diverse types of data provides better understanding of the research problem:

assessing the strengths and shortcomings of a serious game framework.

In SGSEAM, the concurrent triangulation strategy described in Creswell's mixed method approach is used. Data collection and analysis involves both quantitative information (instrument and analytical data recorded by the system such as website logs, interaction database, etc), as well as qualitative information (interviews and questionnaire responses).

SGSEAM shares much in common with the “Goal-Question-Metric” (GQM) approach [?] in software engineering research. GQM defines a software measurement model on three levels: a goal of the measurement, a set of questions to assess the goal, and a set of metrics associated with each question. As there are many metrics related to user experiences [?], SGSEAM focuses on the metrics that are useful to provide insights about the strengths and weaknesses of a serious game framework.

In SGSEAM, the assessment goals are the experiences of the identified stakeholders. For each stakeholder, a set of questions is used to assess the strengths and shortcomings from the stakeholder's perspective. For each question, a set of alternative assessment approaches are described.

## 4.3 Plan the Assessment

This is the first step of SGSEAM. It first identifies the stakeholders, determines the appropriate assessment approaches according to the available resources, and creates the assessment schedule. The deliverable for this step is the assessment plan document which includes the details of stakeholders, approaches and schedule.

### 4.3.1 Identify stakeholders

SGSEAM assesses the experiences for the stakeholders listed in [Table 4.1](#).

For each stakeholder, identify the population, the name and contact if possible. For example, the player stakeholder can be identified as the users who interact with the game interface, perform certain tasks given by the interface, or win a prize. The system admins install, backup, and monitor the software system. The game designers create the content for the game and decide which game to use. The game managers manage the game during the game period. Finally the game developers

Stakeholder class	Definition	Examples
Player	participate in the game produced by the framework.	students, residents
System admin	install and maintain the technological game infrastructure.	system admin, IT staffs
Game designer	design the content and game mechanics.	instructional designers, content experts
Game manager	manage the game during the period of game play.	sustainability coordinators, residential staffs
Game developer	develop customization, extend and enhance the game.	programmers, internal developers

Table 4.1: SGSEAM Stakeholders

develop enhancements and perform customization using APIs provided by the framework.

It is important to be able to contact the stakeholders in some way, either via email or phone, to get the feedback on their experiences with the framework.

#### 4.3.2 Determine assessment approach

There are usually multiple assessment approaches for each stakeholder. [Table 4.2](#) provides an overview of the assessment method and the approaches. The appropriate assessment approaches should be determined according to the resources available. The approaches for a stakeholder are assumed to be additive. The more approaches applied, the higher one's confidence in the accuracy of the assessment results.

The assessment approaches are divided into *in-vivo* and *in-vitro* assessments. The *in-vivo* approaches, such as pre-post test, in-game surveys and post-hoc interviews, assess a real world instance of the game. The *in-vitro* approaches use in-lab experiments in a simulated environment. Different assessment approaches will have different levels of rigor or validity. For example, the in-lab experiments (*in-vitro*) can enlist several subjects to perform the same pre-defined tasks and collect comparable data in a more controlled setting. It is rigorous because of the generality achieved from the larger population of participants under study. On the other hand, in-game surveys or interviews in the *in-vivo* approach typically collect data from different uncontrolled settings with less rigor. But the *in-vivo* data reflect the real world interaction between the stakeholders and the framework, thus providing better insight in the real world settings.

Stakeholder	Assessment goal	Assessment approaches
Player	Determine the extent the framework affect and engage players.	Pre-post effectiveness study(4.3.2.1.1); Self-reported usability survey(4.3.2.1.2); Engagement metrics(4.3.2.1.3)
System admin	Determine strengths and weaknesses in system install and maintenance.	Post-hoc admin interview(4.3.2.2.1); In-lab system admin study(4.3.2.2.2)
Game designer	Determine strengths and weaknesses in facilitating the game design process.	Post-hoc designer interview(4.3.2.3.1); Game design log data analysis(4.3.2.3.3); In-lab game design study(4.3.2.3.2)
Game manager	Determine strengths and weaknesses in managing the game.	Post-hoc manager interview(4.3.2.4.1); Management log data analysis(4.3.2.4.3); In-lab game management study(4.3.2.4.2)
Game developer	Determine strengths and weaknesses in developing system enhancements.	Post-hoc developer interview(4.3.2.5.1); In-lab game development study(4.3.2.5.2)

Table 4.2: SGSEAM approaches

The following sections describe the different approaches for each stakeholder. Each assessment approach describes the goal of the assessment, what data to collect, how to collect the data and how to analyze the data to obtain insights about the strengths and weaknesses of the framework from each stakeholder’s perspective.

#### 4.3.2.1 Player Assessment

The goal of player assessment is to determine the effectiveness of the game framework from the player’s perspective. It is essential that a game produced by a serious game framework achieves its intended “serious” purpose. The intended purposes of serious games are always subject specific. For example, the desired effect of a serious game for energy education and conservation is to increase players’ energy literacy and reduce their energy consumption during (and, hopefully, after) the game. A serious game for language learning would have a very different desired effect.

##### 4.3.2.1.1 Pre-Post effectiveness study

We use a quasi-experimental pre-post study to assess the question of the effectiveness of a serious game framework.

This approach requires users of SGSEAM to first determine a set of domain-specific questions to assess the desired effects of their serious game. For example, a set of questionnaires on sustainability literacy, such as knowledge of power and energy, is used to assess the effectiveness of a serious game for sustainability education.

Once the domain-specific questionnaires are determined and designed, present this questionnaires as a survey to a random selection of the players before the game starts. After the game ends, present the same survey to the same players again. Compare the two sets of survey response data to study if the game has an impact on the players regarding the survey subjects. The extent of the changes reflected in the survey result indicates the degree of effectiveness of the serious game for this subject.

Serious games often engage players with resources of various types (energy, water, waste, etc.). Collect these measurements before, during, and after the game in order to acquire evidence regarding the potential impact upon player use of these resources.

#### **4.3.2.1.2 Self-reported usability survey**

This approach interviews players about player's self-reported experience with the game. We can administer the interview through online surveys or face-to-face conversations, although online surveys are typically more cost effective than face-to-face conversation. If possible, implement the online survey as an activity inside the game. For example, the Makahiki serious game framework implements an online survey activity which incentivizes players to complete the survey by rewarding game points for the activity.

SGSEAM provides a generic set of usability questions outlined in [Figure 4.2](#) that can be used in a usability survey:

#### **4.3.2.1.3 Engagement metrics**

This approach calculates engagement metrics to assess the extent of engagement from players and the impact of the game. The more engaging the game is, the more potential impact it can have on

1. What did you like most about the game?
2. What did you find confusing?
3. What issues did you have while using the game?
4. What was the thing you liked the least about the game?
5. What can we do to improve the game?
6. It was easy to find what I was looking for on the website.  
Strongly disagree - Disagree - Neutral - Agree - Strongly agree
7. The website was responsive.  
Strongly disagree - Disagree - Neutral - Agree - Strongly agree
8. The website provided adequate help in teaching me how to play.  
Strongly disagree - Disagree - Neutral - Agree - Strongly agree
9. I understood how to play.  
Strongly disagree - Disagree - Neutral - Agree - Strongly agree
10. This is something my friends should participate in.  
Strongly disagree - Disagree - Neutral - Agree - Strongly agree

Figure 4.2: Player self-reported usability metrics questionnaires

the players.

Player engagement is an important measure for understanding the effectiveness of a serious game. By investigating the degree of engagement, we can determine to what extent individuals are participating in the game, as well as to what extent the community population is participating in the game. On the other hand, engagement has a subtle relationship to the overall effectiveness of a serious game. It is possible for the game to be played by only a subset of the target population, but have an impact on those not playing by virtue of their social interaction with actual players. Gaining better insight into this diffusion effect is an interesting future direction for research in serious games.

SGSEAM is designed to calculate the player engagement metrics described in [Figure 4.3](#) by analyzing the data from system log or other channels provided by the framework. The more metrics obtained, the better understanding of the extent of player engagement.

The participation rate measures the percentage of users who use the game based on the total number of eligible players. In the serious game context, it indicates the level of involvement or awareness of the issues of interest. The number of players and play time per day measures how frequently the players interact with the game. The submissions per day measures the rate of serious game specific activities (online or real world) that players completed, while the social interaction per

Metric	Definition	Measure
participation	percentage of players who play the game	the level of involvement from players
player	number of players per day	the frequency of players interact with the game
play time	play time of a player per day	the frequency of players interact with the game
submission	submissions of all players per day	the rate of players' completion of game activities
social interaction	social interaction of all players per day	the rate of in-game social interactions between players
game error	game errors per day	the rate of errors encountered by players during the game

Figure 4.3: Player engagement metrics

day measures the rate of social interactions that happened in the game between the players. Finally, the website errors per day measures the rate of errors encountered by the players while using the game website.

With the exception of the game error metric, the higher value these metrics are, the higher engagement level the game has.

#### 4.3.2.2 System admin assessment

System administrators are responsible for installing and maintaining the software infrastructure for the game. Their tasks include the framework and dependency installation, maintaining the database, backups, and so forth. The goal of system admin assessment is to determine to what extent the framework facilitates the system administration tasks from the system admin's perspective. SGSEAM assesses how much time is required to install and maintain an instance of a serious game using the framework and the problems encountered during the system admin process.

SGSEAM proposes two assessment approaches.

##### 4.3.2.2.1 Post-hoc admin interview

This approach assesses the system admin's experience using the post-hoc interview. The system admins are asked about their experience with the framework after they complete the installation and

maintenance of the production system. The interview questions are described in [Figure 4.4](#).

1. How much time did you require to install the system and the dependencies?
2. What problems did you encounter when installing the system and the dependencies?
3. How much time did you require to maintain the system?
4. What problems did you encounter when maintaining the system?
5. Did you find it difficult to admin the system? What was difficult?

Figure 4.4: System admin interview questionnaires

The interview should be tape-recorded. Once the interview is completed, qualitative data analysis is performed against the interview data by doing: (1) transcribing the recordings; (2) coding (categorizing) the time and problems or difficulties encountered. These data reveal the strengths, weaknesses and the areas of improvement for the framework.

#### **4.3.2.2.2 In-lab system admin study**

This approach assesses the system admin's experience using the in-lab experimental study. First identify a group of participants who have some level of system administration experience. Second, provide instruction on each installation step, ask the participants to install the system according to the instructions, and ask them to record the time spent and problems encountered as they complete each step.

Once the experiment data is collected, categorize the reported problems and correlate with the reported time data to identify the areas of strength (less time spent) and weakness (more time spent and problems or difficulties).

The level of confidence of the above two assessment approaches varies. The experimental study approach is more rigorous because of the generality achieved from the larger population of participants under study. The data collected during the step by step experimental study is more accurate than the one collected in the post-hoc interview.

#### **4.3.2.3 Game designer assessment**

A game designer uses the serious game framework to design and create a serious game. A serious game framework normally provides tools or interfaces for game designers to facilitate the design

of a game. For example, the framework provides an interface to configure the game period, set up players, and tools to design individual game elements.

The goal of SGSEAM game designer assessment is to determine the strengths and weaknesses of the framework regarding the game design process. SGSEAM assesses the game designer stakeholder by addressing the following two questions: (a) How much time is required to design an instance of a serious game using the framework? and (b) How many, and how problematic are the errors that designers encounter during the design process?

There are two approaches for game designer assessment:

#### **4.3.2.3.1 Post-hoc designer interview**

This approach interviews the game designer(s) after they complete the design of a serious game using the framework in a production system. The interview includes the questions described in [Figure 4.5](#).

1. How much time did you spend to complete each design task?
2. What problems did you encounter?
3. Did you find it difficult to configure? What was difficult?
4. Did you find it difficult to design a specific game? Which one, and what was difficult?

Figure 4.5: Game designer interview questionnaires

The interview should be tape-recorded. After the interview, transcribe the recordings, code and categorize the reported time and problems to identify the strengths and weaknesses.

In addition, if possible, collect system log data related to the game designing tasks, and analyze the logs to find out the time spent and the errors encountered during the game designing tasks. Use the log data to verify the findings from the interview data.

#### **4.3.2.3.2 In-lab game design study**

This approach assesses the game designer experience using an in-lab experimental study. First identify a group of participants who are at least somewhat familiar with the subject domain of the game. Second, provide instructions on each design step, ask the participants to design the game

according to the instructions, ask them to record the time spent and problems encountered as they complete each step.

Once the experimental data is collected, categorize the reported problems and correlate with the reported time data to identify the areas of strength (less time spent) and weakness (more time spent and problems or difficulties).

#### **4.3.2.3.3 Game design log data analysis**

This approach collects the system log data related to the game designing tasks. When available, the time spent and errors encountered can be queried from the system logs. Although this system generated data might be easier to gather in some systems, it might not provide the same depth of insight as the other two approaches where the experiences are provided by the participants directly. On the other hand, this system data can be supplemental to the other approaches. They can be correlated with the data gathered from the other assessment approaches to increase the confidence of the assessment.

#### **4.3.2.4 Game manager assessment**

A game manager uses the serious game framework interface to manage the serious game that the game designers created. It is possible that a game manager is also the game designer. Examples of game management tasks includes managing player submissions, monitoring the game state, entering manual resource data, notifying winners of the game, etc.

The goal of SGSEAM game manager assessment is to determine the strengths and weakness of the framework regarding the game management process. Similar to the assessment of the game designer interface, SGSEAM assesses the game manager interface regarding the time required to manage an instance of a serious game using the framework and the problems encountered during the management process.

SGSEAM proposes two approaches for assessing the game manager's experience.

#### **4.3.2.4.1 Post-hoc manager interview**

This approach interviews the game manager(s) after they have finished managing a serious game using the framework in a production environment. The interview questions are described in [Figure 4.6](#).

1. How much time did you spend to complete each managing task?
2. What problems did you encounter?
3. Did you find it difficult to manage? What was difficult?

Figure 4.6: Game manager interview questionnaires

The interview should be tape-recorded. After the interview, transcribe the recordings, code and categorize the reported time and problems to identify the strengths and weaknesses.

In addition, if possible, collect the system log data related to the game management tasks, analyze the logs to find out the time spent and errors encountered during the game managing tasks. Use the log data to verify the findings from the interview data.

#### **4.3.2.4.2 In-lab game management study**

This approach assesses the game manager's experience using the in-lab game management study. First identify a group of participants who are somewhat familiar with the subject domain of the game. Second, provide instructions on each managing tasks, ask the participants to complete the tasks following the instructions, ask them to record the time spent and problems encountered as they complete each task.

Once the experiment data is collected, categorize the reported problems and correlate with the reported time data to identify the areas of strength (less time spent) and weakness (more time spent and problems or difficulties).

#### **4.3.2.4.3 Game management log data analysis**

This approach collects and analyzes the system log data related to the game managing tasks. The time spent and errors encountered can be deduced from the system log and reveals strengths and

weaknesses of the game management interface.

#### **4.3.2.5 Game developer assessment**

The game developer stakeholder is different from the game designer stakeholder, in that the game designer stakeholder tailors the framework without requiring any software development, while the game developer stakeholder enhances, corrects, and extends the system by manipulating code.

To investigate how easy it is to understand, extend, and debug a serious game framework from a developer's perspective, SGSEAM assesses how much time it takes to develop an enhancement to the game framework, and how many errors are encountered during the development process.

##### **4.3.2.5.1 Post-hoc developer interview**

This approach interviews the game developer(s) to assess their experiences in developing the game using the framework. The interview questions are described in [Figure 4.7](#).

1. How much time did you spend developing a customization using the game framework?
2. What problem(s) did you encounter?
3. Did you find it difficult to understand, extend and debug the system? What was difficult?

Figure 4.7: Game developer interview questionnaires

##### **4.3.2.5.2 In-lab game development study**

This approach assess the game developer's experience using an in-lab game development study. First identify the general development skills that the framework requires, such as the programming language. Second, identify a group of participants who possess the required development skills. Third, provide a software requirements specification or instructions on how to develop a new enhancement to the system, ask the participants to complete the task, record the time spent and problems encountered as they work on the task.

Once the experimental data is collected, categorize the reported problems and correlate with the reported time data to identify the areas of strength (less time spent) and weakness (more time spent

and problems or difficulties).

### **4.3.3 Choose participants**

Once the assessment approaches are determined for each stakeholder class, the next step is to choose participants. Identify the people from the each stakeholder class that may be willing to participate in the assessment, contact them and get consent for their participation.

For example, in the case of the pre-post effectiveness study approach for player assessment, this step randomly chooses a group of players and presents a consent form before the online survey. In the case of the post-hoc game designer interview approach, the game designer of a real world game instance of the framework should be identified, contacted and consent obtained for participating in the assessment. When the in-lab game development experiment study is chosen, a group of game developers that meet the required development skills of the framework should be identified and recruited.

### **4.3.4 Create assessment schedule**

After choosing the assessment approaches and the participants, the next step is to create the assessment schedule. This document should include a detailed assessment plan for each stakeholder class.

Depending on the assessment approach, the actual tasks of the assessment are different. The player pre-post effectiveness study requires the administration of an online survey before and after the game. The game designer post-hoc interviews requires administration of interviews to the real world game designers of a production system. [Table 4.3](#) shows an example of the assessment schedule broken down by the tasks in the plan document.

## **4.4 Gather Data**

Once the plan has been finalized, the next step is to carry out the assessment, record the data, obtain log data, and (if necessary) refine the assessment plan. The output of this step is a data repository containing all the assessment data that can be analyzed in the next step.

Game design assessment approach: in-lab experiment study		
Task	Estimated Start date	Estimated End date
Design the in-lab experiment instruction		
Ask participants to follow the instruction		
Collect response data from participants		
Obtain log data		
Analyze the data		
Interpret strength and weakness		
Produce action document		

Table 4.3: Assessment schedule in the plan document

#### 4.4.1 Carry out the assessment

For each assessment approach, complete the tasks outlined in the assessment plan and gather the data when carrying out the assessment. In the case of the game designer post-hoc interview approach, record the interview and take notes if necessary. Store all the data in a central data repository.

In the example of the in-lab game design experiment study, a google form can be used to give detailed step by step instructions for the participants to design games using the framework. Participants can be asked to record the time they spent completing each step and the problems they encountered. They can also be asked to provide feedback about their design experiences in written form.

#### 4.4.2 Obtain log data

Talk to the technical staff of the framework to find out what kind of log data is available. Obtain the log data in a format that is easy to analyze. For example, if the log data is in a database table, ask for access to the table, or the CSV export of the table data. If the log data is in a log file, ask for access to the file. Store the log data into the central data repository.

### 4.5 Produce Assessment Report

In this step, analyze the data gathered from previous steps, create an analysis of the strengths and weakness of the framework, and produce an action report with your recommendations as to frame-

work improvements.

#### **4.5.1 Analyze Data**

This step performs the data analysis from the data repository obtained from the previous step. For the game designer assessment, perform queries from the designer interaction log data to find out the completion time for each design task, for instance, the time for a game designer to complete the configuration of global game settings. For player interaction assessment, calculate the engagement metrics from the game log. For the post-hoc interview assessment approach, first transcribe the interview recording into text, code and categorize the responses from the interview questions.

In the example of the in-lab game design experiment study described previously, the assessment data is generalized into 7 tasks corresponding to distinct types of game design tasks. The time for each task is calculated from the Google form responses. The problems reported from the participants are coded and aggregated into problems areas.

#### **4.5.2 Determine strength and weakness**

This step determines the most important problem areas from our data and summarizes them, as well as the areas where the framework appears to be most successful.

In the example of the in-lab design experiment study, there may be a problem area that had been reported by the most numbers of participants, and if this problem happened in one of the tasks that took the longest time to complete, we can identify a weakness area of the framework from the perspective of a game designer. If there were no problems reported in a game design task and the time to complete is short, we can consider those areas to be the strengths of the framework.

#### **4.5.3 Produce Report with Actionable Steps**

Once the strengths and weaknesses of the framework are identified, an action report should be produced. This report includes the weakness areas that can be improved and actionable steps on how to improve from each stakeholder's perspective. It also includes strengths that the framework needs to maintain.

**Table 4.4** shows a template for the improvement action report.

Stakeholder class	Strength	Weakness	Improvement action
Player			
System admin			
Game designer			
Game manager			
Game developer			

Table 4.4: SGSEAM Improvement Action Report Template

By producing the report with actionable steps to improve the framework, the SGSEAM assessment is completed.

## 4.6 Summary

In summary, SGSEAM describes a method for assessing serious game frameworks from the stakeholder experience perspective, namely the five types of stakeholders who are affected by serious game frameworks: players, system admins, game designers, game managers, and game developers.

The SGSEAM process consists of three steps. The first step is to plan the assessment to identify the stakeholder and determine the assessment approaches. The assessment approaches are divided into in-vivo and in-vitro assessments. The in-vivo approaches, such as pre-post test, in-game surveys and post-hoc interviews, assess a real world game instance. The in-vitro approaches use in-lab experiments in a simulated environment. For each stakeholder, multiple assessment approaches can be used depending on the resources available. The more approaches applied, the higher one's confidence in the accuracy of the assessment results. The second step in the SGSEAM process is to gather the data by carrying out the assessments described in the plan. The last step in the SGSEAM process is to analyze the data and produce the assessment report.

## CHAPTER 5

# MAKAHIKI AND SGSEAM EVALUATION

This chapter describes the way I evaluated the Makahiki framework described in [Chapter 3](#) and the SGSEAM method described in [Chapter 4](#). First, I describe the real-world case studies of the Makahiki instances realized in the Kukui Cup Challenges at the different organizations, followed by the detailed assessment of applying SGSEAM to Makahiki framework. The evaluation is to address: (a) obtain insights about the strength and weakness of the Makahiki serious game framework, (b) obtain insights about the strength and weakness of the SGSEAM serious game framework assessment method.

### **5.1 Real-world Makahiki Instances Case Studies**

Makahiki, as a serious game framework for sustainability, has been used by different organizations to create multiple serious game instances targeting to education and fostering sustainable behavior in communities. The first Kukui Cup Energy challenge at the University of Hawaii at Manoa (UHM) was held in 2011 for 3 weeks for over 1,000 first year students living in the residence halls. UHM subsequently held the second and third Kukui Cup Energy challenges in 2012 and 2014 for different first year students and different durations, for 9 months and 2 weeks respectively. Hawaii Pacific University (HPU) held their Kukui Cup Energy challenges in Fall 2012 and 2013 for about 200 students each year. An international organization called the East-West Center (EWC) held a Kukui Cup Energy and Water challenge for the international residents living in the residence halls. A Hawaii private school called Holy Nativity School (HNS) held a pilot Kukui Cup challenge for their elementary school students.

[Table 5.1](#) lists these instances and their different requirements. The major differences involve the duration of the challenge, the population that could participate in the challenge, the type of resource(s) such as energy or water, whether they have smart meters installed, and the type of server hosting. Additional differences in requirements include the type of the authentication for participation and differences in game mechanics. These differences will be described in the results

chapter in more detail.

Instances	Duration	Participants	Resource	Smart meters	Hosting
UHM 2011	3 weeks	1038	Energy	✓	Local
UHM 2012	9 months	1067	Energy	✓	Cloud
UHM 2014	2 weeks	1056	Energy	✓	Cloud
HPU 2012	3 weeks	198	Energy	✓	Local
HPU 2013	3 weeks	197	Energy	✓	Local
EWC 2012	2 weeks	129	Energy & Water		Cloud
HNS 2013	7 months	10			Cloud

Table 5.1: Makahiki Serious Game Instances

The Makahiki instances deployed at these organizations have to meet these different requirements. The goal of the Makahiki framework is to minimize the effort needed to support the different requirements in various organizations. The real world case study of Makahiki looks at the different requirements of these organizations, and how easily the Makahiki framework can be configured to support these requirements, thus gaining insight into how well the Makahiki framework can be used and customized in real world scenarios. The results of the real world case studies of these instances are described in [Section 6.1](#) of the Results chapter.

## 5.2 Applying SGSEAM to Makahiki

This section describes a formal assessment of the strengths and weaknesses of the Makahiki framework by applying SGSEAM to Makahiki. As described in [Chapter 4](#), SGSEAM is a method to assess a serious game framework based on different serious game stakeholders' experiences. I applied SGSEAM to the Makahiki framework to identify the strengths and weaknesses of both the Makahiki and the SGSEAM itself.

### 5.2.1 Step 1: Plan the Assessment

The first step of the SGSEAM process is to plan the assessment. The deliverable for this step is the assessment plan document. To complete this step, I created a tech report called “SGSEAM Assessment Plan for Makahiki”[?], also included in [Appendix B](#). The assessment plan identifies

the stakeholders, determines the appropriate assessment approaches according to the available resources, chooses assessment participants, and creates the assessment schedule.

The SGSEAM assessment plan includes two categories of assessment approaches for Makahiki: the in-vivo approaches applied to the real-world UHM, HPU and EWC Makahiki instances, and the in-vitro approaches applied to the UHM ICS691 serious game course as an in-lab experiment.

[Table 5.2](#) provides an overview of assessment plan for each SGSEAM stakeholders of the Makahiki framework. [Appendix B](#) provides the detailed descriptions of the plan.

Stakeholder	Assessment Approach	Participants	Time
Player	Pre Post effectiveness study Self-reported effectiveness survey Self-reported usability survey Engagement metrics	UHM Kukui Cup players	2011, 2012, 2014
System admin	Post-hoc system admin interview	HPU KC sysadmin	2012
	In-lab installation study	UHM ICS691 students	2013
Game designer	Post-hoc game designer interview	HPU & EWC KC game designers	2012
	In-lab game design study	ICS691 students	2013
Game manager	Post-hoc game manager interview	HPU & EWC KC game managers	2012
Developer	In-lab game development study	ICS691 students	2013

Table 5.2: SGSEAM Assessment Plan Overview for Makahiki

### 5.2.2 Step 2: Gather Data

The second step of the SGSEAM process is to gather the assessment data by carrying out the assessment described in the assessment plan, recording the data and obtaining game logs. The output of this step is a data repository containing all the assessment data.

During the period of 4 years from 2011 to 2014, the SGSEAM assessment for Makahiki was carried out by implementing different assessment approaches for the stakeholder participants, both in real-world Makahiki instances and in-lab experiments. I collected the necessary assessment data and stored them in a central location in a hard drive. [Appendix C](#) describes the content of the data repository collected from the SGSEAM assessment of Makahiki.

The following sections describe the assessments and the data collected.

### **5.2.2.1 Player Assessment**

We used the real-world Makahiki instances at the University of Hawaii at Manoa to assess the players' experience with Makahiki.

To assess the effectiveness of the framework for designing games that improve player literacy in sustainability, we conducted two energy literacy surveys, one before the challenge (pre-game) and one after the challenge (post-game). SurveyGizmo was used to create the surveys which consists of the set of sustainability literacy and behavior questionnaires. After the surveys were completed, the responses saved in SurveyGizmo were exported to be stored in the SGSEAM assessment data repository.

To assess the effectiveness of the framework for designing games that produce positive change in sustainability behaviors, we recorded and analyzed energy consumption data before, during and after the challenge, and stored the energy consumption data into the SGEEAM assessment data repository.

We also conducted the in-game self-reported behavior changes survey. The survey asked questions about player interests in sustainability prior to and after the game, as well as any perceived behavior changes when playing the game. To assess the usability of the game produced by the Makahiki framework, we conducted an in-game usability survey. The survey asked questions about the players' experience with respect to the user interface of the game. The surveys were administered using SurveyGizmo service. The responses were exported to be stored in the SGSEAM data repository.

In addition to the surveys and energy data, we collected the game and log data to calculate the engagement metrics to assess the engagement level of the instance. The game data which were saved in the Makahiki database were exported and stored in the SGSEAM data repository.

The following lists the data collected with regards to the SGSEAM player assessment:

- Pre-post surveys from players;
- Energy data before, during, and after the game;
- In-game survey from players;

- Game data and logs (participation and submissions)

[Appendix C](#) describes the content of these data in the data repository.

### 5.2.2.2 System Admin Assessment

There are two approaches described in SGSEAM to assess the system admin's experience: One is the interview of the system admin of a real world instance, the other is the in-lab experiment with the ICS691 serious game development class.

In order to gain insights into the experience of a real world system admin who uses Makahiki, I performed interviews with the system admin of the Hawaii Pacific University (HPU) challenge. I also collected email exchanges with the system admin regarding the installation process and maintenance of Makahiki system. Both interview audio recordings and email exchange data were stored in the SGSEAM data repository.

In the in-lab experiment, the students in the ICS691 Spring 2013 class were tasked with installing the Makahiki system into their local computers as well as the cloud environment. In order to understand how much time it takes to install the Makahiki and what problems might be encountered, a Google form was used to describe the steps for installing Makahiki both locally and in the cloud. For each step, students were asked to record the time they spent and the problems they encountered. [Appendix G](#) includes the complete installation Google form used for Makahiki system admin assessment.

The students were also asked to provide feedback about their installation experiences in the form of blog posts. The responses from the Google form and the blog posts were exported and stored in the SGSEAM data repository.

The following lists the data collected with regards to the SGSEAM system admin assessment:

- Interview audio recordings with system admins;
- Email communications with system admins;
- Google form responses from the ICS691 installation experiment;

- Blog posts from ICS691 installation experiment

[Appendix C](#) describes the content of these data in the data repository.

### 5.2.2.3 Game Designer Assessment

Similar to the SGSEAM system admin assessment, there are also two approaches described in the SGSEAM plan to assess the game designer's experience: interviews and in-lab experiment.

I performed the interviews with the real-world game designers of the Hawaii Pacific University challenge and the East-West Center challenge. I asked them about their game designing experiences using the Makahiki game design admin interface. The email exchanges with the game designers regarding the game design process were also collected. Both interview audio recordings and email exchange data were stored in the SGSEAM data repository.

The students in the in-lab experiment were tasked to design a Kukui Cup-like serious game using Makahiki. I designed another Google form to ask students to follow the design steps and record their time and problems encountered during their design process. [Appendix G](#) has the complete game design Google form for the steps the students needed to follow. The students were also asked to provide feedback about their installation experiences in the form of blog posts that discussed the difficulties and problems they encountered during the game design process.

The data collected from the Google form response and the blog post from the students were analyzed. The results are described in [Section 6.2.3.1](#) of the Results chapter.

The responses from the Google form and the blog posts were exported and stored in the SGSEAM data repository.

The following lists the data collected with regards to the SGSEAM game designer assessment:

- Interview audio recordings with game designers;
- Email communications with game designers;
- Google form responses from ICS691 game design experiment;
- Blog posts from ICS691 game design experiment

[Appendix C](#) describes the content of these data in the data repository.

#### **5.2.2.4 Game Manager Assessment**

I performed the interviews with the game managers of the Hawaii Pacific University challenge and the East West Center challenge to study the experience of game management using Makahiki. I also collected the email exchanges with the game managers regarding the game management process. Both interview audio recordings and email exchange data were stored in the SGSEAM data repository.

The following lists the data collected with regards to the SGSEAM game designer assessment:

- Interview audio recordings with game managers;
- Email communications with game managers;

[Appendix C](#) describes the content of these data in the data repository.

#### **5.2.2.5 Developer Assessment**

I performed the in-lab game development experiment with the students participating in the ICS691 serious game development class. The students in the experiment were tasked with developing an enhancement to the Makahiki instance. This involved setting up the development environment, following the tutorial to create the “Hello world” widget using Makahiki, and finally, developing an enhancement which extends the functionality of the Makahiki system. The students were asked to submit their development source code to the public source code repository Github and write a blog post to discuss their efforts to complete the development activity.

The blog posts were exported and stored in the SGSEAM data repository. The following lists the data collected with regards to the SGSEAM game designer assessment:

- Blog posts from ICS691 game design experiment

### **5.2.3 Step 3: Produce Assessment Report**

After the assessment approaches were carried out and data collected, the data was analyzed to determine the strengths and weaknesses of the Makahiki framework. The results of the analysis are described in the [Section 6.2](#) of the following Results chapter. The final deliverable of the SGSEAM assessment process was created as the improvement action report described in [Appendix D](#).

## **5.3 Consent from Human Research Study Subjects**

This research interviewed and studied the behaviors of several human subjects including:

- First year students who played the real world UHM Kukui Cup instances for 2011, 2012 and 2014
- Students who participated in the UHM ICS691 in-lab experiment study
- Administrators who used Makahiki to install, design, and manage the real world Kukui Cup instances in HPU and EWC

We had obtained their consent to participate into this research from all the above human subjects. The study is also approved by Office of Research Compliance at the University of Hawaii Human Studies Program under the CHS #20451.

## **5.4 SGSEAM Assessment plan for Lucid BuildingOS**

With the hope to collaborate with Lucid Design Group to apply SGSEAM to their Building OS and Building Dashboard serious game framework, I created the SGSEAM assessment plan for them. The plan is described in [Appendix E](#) as part of the proposed SGSEAM assessment guide for Lucid BuildingOS.

[Table E.3](#) shows the approaches we recommended for each stakeholder in the case of Lucid BuildingOS framework.

Stakeholder	Assessment Approaches	Expected Outcomes	Comments on approach
Player	Pre-post effectiveness study	Determine effectiveness in resource usage reduction.	
	Usability survey	Identify problem areas in game interface	
	Engagement metrics	Determine the extent of engagement	
System admin	Post-hoc admin interview	Identify strengths and weaknesses in the installation and maintenance process.	
Game designer	Post-hoc designer interview	Determine strengths and weaknesses in the game design interface.	
Game manager	Post-hoc manager interview	Determine strengths and weaknesses in the game managing interface.	
Game developer	Post-hoc developer interview	Determine strengths and weaknesses in developing enhancement.	

Table 5.3: BuildingOS Assessment Approaches

Comparing to the approaches of the SGSEAM assessment of Makahiki, the recommendation mainly consists of *in-vivo* approaches, such as post-hoc interviews, pre-post test, and in-game surveys. While an in-lab experiment such as the one done in Makahiki is more rigorous, we believe it is too expensive for Lucid’s assessment. On the other hand, the in-vivo data reflect the real world interaction between the stakeholders and the framework, thus providing better insight in the real world settings.

By carrying out the assessment described in the plan, we hope to identify the major strengths and weaknesses of the Lucid’s BuildingOS framework from the perspectives of major stakeholders, and provide recommendations to any actionable improvements to the developers of the framework.

## 5.5 Limitations

There are several limitations to this evaluation design. SGSEAM is designed to assess any serious game framework from the five major serious game stakeholder’s perspectives. We only applied SGSEAM to one serious game framework, Makahiki, which was also designed by us. I have biases resulting from my experience implementing Makahiki that may affect the design of the SGSEAM, as a result of which some of the assessment approaches might not be effective in assessing other serious game framework. As described in [Section 7.3.1](#) of the “Future Directions” section, BuildingOS[?] by Lucid Design Group, another serious game framework, is a good candidate to be assessed with SGSEAM. Our research lab made an effort to contact Lucid Design group for collaboration. I

created the assessment plan ([Appendix E](#)) for them which we hoped would lead to the application of SGSEAM to the BuildingOS system. But due to the workload of the Lucid design group, the collaboration did not continue.

Without another data point of the application of SGSEAM, there is not enough confidence in the evaluation of SGSEAM itself.

One limitation in the application of SGSEAM to Makahiki is the small sample size in the post-hoc interview approach. Due to the nature of the this kind of serious game, it is common to have just one or two system admin(s), game designer(s), and game managers(s). It is often the case that the game designers are the same as the game managers. In our evaluation design, although there are 7 Makahiki game instances in 4 organizations, we can only interview 1 system admin from the HPU instance and 3 game designers who are also game managers from the HPU and EWC instances. The UHM instances were designed and managed by ourselves. We also installed and deployed the instances except the HPU ones, so there is no other sys admin candidate.

Although the in-lab experiment study is a good approach to overcome the small sample size limit of the post-hoc interview approach, the ICS691 students participating in the system admin experiment might not reflect the typical system admin population because of the different skill set, thus limiting the accuracy of the assessment results. On the other hand, it would be difficult or expensive to choose a group of participants with system admin skills.

Given these limitations, the application of SGSEAM to Makahiki still resulted in useful insights into the strengths and weaknesses of the Makahiki framework.

## 5.6 Summary

This chapter described the evaluation design for both Makahiki and SGSEAM. Makahiki, as a serious game framework for sustainability, has been used by different organizations to create multiple serious game instances targeting education and fostering of sustainable behavior in communities. The real world case study of Makahiki looks at the different requirements of these organizations and the corresponding different configurations in the Makahiki framework to support these different requirements. A more formal assessment of the strength and weakness of the Makahiki framework is

to apply SGSEAM to Makahiki.

Following the SGSEAM process, the assessment plan ([Appendix B](#)) was created as the first deliverable of the process, also as a technical report [?] to guide the SGSEAM assessment of Makahiki. The assessment plan identifies the participants in the categories of five SGSEAM stakeholders. Two categories of assessment approaches were determined. The *in-vivo* assessment approaches of pre-post effectiveness study, in-game surveys and post-hoc interviews were applied to the real-world Kukui Cup challenges created by the Makahiki framework, to assess the experiences of players, system admins, game designers, and game managers. The *in-vitro* assessment approaches of in-lab study used the UHM ICS691 serious game course to assess the experiences in game installation, game design and game development. The assessment schedule was also created during the planning step.

After creating the assessment plan, I carried out the Makahiki assessment according to the SGSEAM plan and a data repository ([Appendix C](#)) was created to collect all the assessment data. The data was analyzed to determine the strengths and weaknesses of the Makahiki framework. The results of the analysis are described in the [Section 6.2](#) of the Results chapter. The final deliverable of the SGSEAM assessment process was created as the improvement action report described in [Appendix D](#).

# **CHAPTER 6**

## **RESULTS**

This chapter reports the results of several real world instances of Makahiki implemented in different organizations, as well as the application of SGSEAM to the Makahiki framework.

### **6.1 Real-World Case Studies**

We used Makahiki to create a total of seven Kukui Cup Energy Challenges in four different organizations. The following sections describe the Makahiki instances for the four organizations and the different configurations for these instances.

#### **6.1.1 Makahiki instances**

##### **6.1.1.1 University of Hawaii at Manoa**

There were three instances of Kukui Cup (KC) Challenges implemented by using the Makahiki framework at the University of Hawaii at Manoa (UHM). They were held in 2011, 2012, and 2014 respectively for over 1000 first year students per year living in the residence halls on campus. The three instances had competition durations of 3 weeks, 9 months and 2 weeks respectively. The residence halls where the students were living had energy smart meters installed for collecting the energy data consumed by the students.

##### **6.1.1.2 Hawaii Pacific University**

There were two instances of Kukui Cup Challenges implemented using the Makahiki framework at Hawaii Pacific University (HPU). They were held in 2012 and 2013 respectively for about 200 students per year living in 6 residence halls at the Hawaii Loa Campus. Energy smart meters were installed in the residence halls.



Figure 6.1: UHM Kukui Cup Challenge Home Page



Figure 6.2: HPU Kukui Cup Challenge Home Page

### 6.1.1.3 East-West Center

The EWC Kukui Cup Energy and Water challenge was implemented by an international organization called the East-West Center (EWC) using the Makahiki framework. It was held in 2012 for approximately 600 international students living in two residence halls in Hawaii. The challenge lasted for 2 weeks and includes energy and water saving competition between two residence halls. The residence halls did not have internet-enabled smart meters, so energy data was collected by program organizers by reading the meters each day at the same time and entering the readings into the system manually.



Figure 6.3: EWC Kukui Cup Challenge Home Page

#### **6.1.1.4 Holy Nativity School**

A pilot instance of Kukui Cup challenge implemented by using the Makahiki framework was held at the Holy Nativity School (HNS), a private elementary school in Hawaii, in 2013. The pilot instance was organized by the school with the partnership with Project Learning Tree (PLT) GreenSchool! program[?]. This nationwide environmental service-learning program helps improve student academic performance in STEM subjects by addressing environmental issues at their school. This challenge was designed so that it did not require any energy or water data. It was used to supplement STEM education with a sustainability focus. The challenge was initially scheduled for one month. It was later extended to 7 months.

### **6.1.2 Customization of the Makahiki Instances**

The following sections describe the different customizations that were done to the above Makahiki instances to address each organization's needs.

#### **6.1.2.1 Challenge Customization**

Challenge customization includes the duration of the challenge, the participant accounts, resource customization such as energy and water settings, learning action configuration, prize and other game mechanics settings.



Figure 6.4: HNS Kukui Cup Challenge Home Page

**Table 6.1** lists the different configurations between the seven real world instances of Makahiki.

Instances	Participants	Teams	Duration	Rounds	Game Element			
					Energy	Water	Prize	Quest
UHM2011	1070	20	3 weeks	3	✓		✓	✓
UHM2012	1086	20	9 months	4	✓		✓	✓
UHM2014	1093	20	2 weeks	2	✓		✓	✓
HPU2012	189	6	3 weeks	3	✓		✓	✓
HPU2013	197	6	3 weeks	3	✓		✓	✓
EWC2012	130	2	2 weeks	1	✓	✓		
HNS2013	10	2	7 months	1			✓	✓

Table 6.1: Challenge Configuration Differences

As we can see from the **Table 6.1**, the Makahiki framework can be customized to support different numbers of participants, different team sizes, different durations, different numbers of rounds, and different types of game elements. For example, while the UHM and HPU challenges involved only energy consumption data, the EWC challenge involved both energy and water consumption data.

### 6.1.2.2 Content Customization

Because different organizations have different sustainability education needs, they used the Makahiki framework to customize the content, which is shown to players via the SmartGrid Game. They can

re-use or modify the existing contents included with the Makahiki framework or create new content to be included in the system. [Table 6.2](#) shows the differences in type and layout of educational contents. UHM had the most number of learning actions while HNS has the least.

Instances	Levels	Activities	Commitments	Events	Total Actions
UHM2011	1	51	21	21	93
UHM2012	7	68	23	35	126
UHM2014	4	60	21	20	101
HPU2012	3	24	11	4	39
HPU2013	3	29	11	4	44
EWC2012	4	21	1	19	41
HNS2013	2	22	6	4	32

Table 6.2: Content Differences

The layout of the educational content represented in the SmartGrid game is also highly customizable to include different levels, rows and columns. The figures [Figure 6.5](#), [Figure 6.6](#), [Figure 6.7](#), [Figure 6.8](#) illustrate the content and layout of the educational SmartGrid game in the different Makahiki instances.

#### 6.1.2.3 Branding Customization

The look and feel of the challenges website are difference between the different organizations, which is customized using the customization feature of the Makahiki framework. The user interface was customized to “brand” each challenge. The list of customizable branding are:

- Logo
- Site name
- Challenge name
- Team label
- Landing page text
- About page text
- Sponsor text and logo



(a) UHM 2011 Kukui Cup SmartGrid Game



(b) UHM 2012 Kukui Cup SmartGrid Game



(c) UHM 2014 Kukui Cup SmartGrid Game

Figure 6.5: UHM SmartGrid Game Layouts



(a) HPU SmartGrid Game Level1



(b) HPU SmartGrid Game Level2

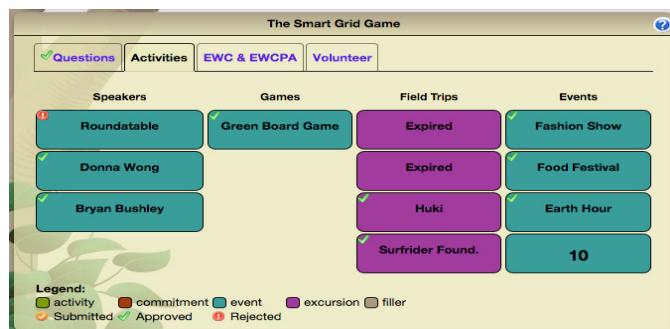


(c) HPU SmartGrid Game Level3

Figure 6.6: HPU SmartGrid Game Layouts



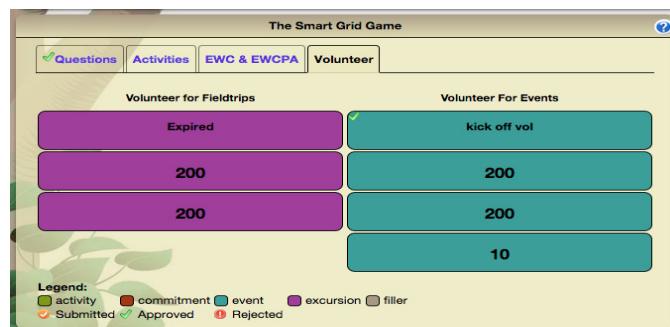
(a) EWC SmartGrid Game Level1



(b) EWC SmartGrid Game Level2



(c) EWC SmartGrid Game Level3

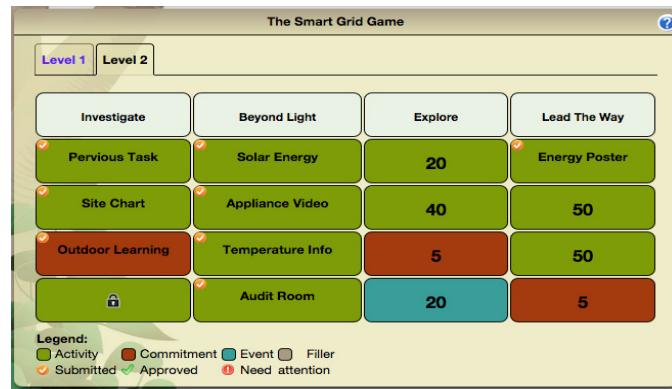


(d) EWC SmartGrid Game Level4

Figure 6.7: EWC SmartGrid Game Layouts



(a) HNS SmartGrid Game Level1



(b) HNS SmartGrid Game Level2

Figure 6.8: HNS SmartGrid Game Layouts

- Theme

#### 6.1.2.4 System Configuration

The system configuration includes the email server integration, authentication server integration, whether to use Facebook or not, and the smart meter connections.

**Table 6.3** lists the different configurations between the seven real world instances of Makahiki.

Instances	Email server	Authentication server	Use Facebook	Smart meters	Smart meter model
UHM2011	Gmail	UHM CAS	✓	✓	Shark
UHM2012	Gmail	UHM CAS	✓	✓	Shark
UHM2014	Gmail	UHM CAS	✓	✓	Shark
HPU2012	HPU email	HPU LDAP	✓	✓	eGauge
HPU2013	HPU email	HPU LDAP	✓	✓	eGauge
EWC2012	Gmail	UHM CAS& Internal	✓		
HNS2013	Gmail	Internal			

Table 6.3: System Configuration Differences

In order to send emails to the players, Makahiki needs to be configured to use an email server to send the emails. In the UHM, EWC and HNS cases, the Gmail server was used to send out emails. In the HPU case, the system admin created a specific email account for the HPU Kukui cup (kukuicup@hpu.edu) in their existing email server, and configured Makahiki to use the HPU email server.

The IT infrastructure at UH and HPU provided authentication services using CAS (Central Authentication Service) and LDAP, while EWC used the built-in Django authentication.

Because the HNS instance was targeted for elementary school, Facebook integration was turned off in its configuration. In the other instances, Facebook integration was turned on and was used extensively during the game.

UH and HPU used different metering infrastructure, and EWC collected their resource data manually. Since the EWC residence halls did not have internet-enabled meters, resource consumption data had to be entered by the game managers manually.

### 6.1.3 Cloud Hosting Experiences

Makahiki supports two types of hosting configurations. One is local hosting where the application runs on the organization's own IT infrastructure. The other is cloud hosting where the application runs on the Heroku's cloud infrastructure.

**Table 6.4** lists the different hosting configurations between the seven real world instances of Makahiki. For the instances that hosted on the cloud infrastructure, the cloud service costs are also included in the table.

Instances	Hosting	Other cloud services	Duration	Cloud service cost
UHM2011	Local Mac OS server	N/A	3 weeks	N/A
UHM2012	Heroku cloud	Amazon S3, Memcache	9 months	\$908
UHM2014	Heroku cloud	Amazon S3, Memcache	2 weeks	\$150
HPU2012	Local CentOS server	N/A	3 weeks	N/A
HPU2013	Local CentOS server	N/A	3 weeks	N/A
EWC2012	Heroku cloud	Amazon S3, Memcache	2 weeks	\$9
HNS2013	Heroku cloud	Amazon S3, Memcache	7 months	\$146

Table 6.4: Hosting Configuration Differences

We installed Makahiki for the UHM 2011 instance locally in a Mac XServe server hosted in the Information and Computer Science department at the University of Hawaii at Manoa. We assisted the installation of the HPU instance by working with the system admin personnel at HPU. For the EWC, HNS and the 2012, 2014 UHM instances, we deployed and managed the instances in the Heroku cloud infrastructure.

Although there are some costs in the case of hosting the Makahiki games in the Heroku cloud infrastructure, there is no need for maintaining the local infrastructure, such as keeping the server running, upgrades, patching as well as locating physical space for the server. There is no need to allocate IT support staff resource to do the system admin. The long term cost of managing the local infrastructure is potentially higher. A serious game normally runs for a certain duration, so the organization only needs to pay for the charge for cloud infrastructure during that period.

The following sections discuss the cloud services Makahiki used and its associated cost:

### 6.1.3.1 Heroku Cloud Service: Dynos and Database

One major benefit of using cloud infrastructure is the capability of dynamically adjust the computing resource for the application with scaling up when the usage demand is high and scaling down when the usage demand is low. This can be easily managed in the Heroku cloud infrastructure by the way of adjusting the numbers of Dyno and different type of database allocated to the application. Dynos are the units of computing power on Heroku, isolated containers that run the deployed application. Heroku provides different sizes (computing power) of Dynos and Postgres databases with different pricing [?]. [Table 6.5](#) lists the resource allocations to the real world instances of Makahiki in the Heroku cloud infrastructure. At the time of the hosting, there was only one size of dynos which was provisioned as 512MB RAM with 1xCPU Share. They were priced as \$34.50 per month, with the first dyno free of charge.

Instances	Avg. Number of Dynos	Database	Avg. Monthly Cost
UHM2012	2.8	Standard 0: 1 GB RAM, 64 GB storage, 120 connections (\$50/month)	\$100
UHM2014	4	Standard 2: 3.5 GB RAM, 256 GB storage, 400 connections (\$200/month)	\$300
EWC2012	1	Basic: 10 million rows limit (\$9/month)	\$10
HNS2013	1.3	Basic: 10 million rows limit (\$9/month)	\$20

Table 6.5: Heroku Hosting Configuration

The UHM2014 instance had the highest monthly cost while the EWC2012 instance had the lowest monthly cost. The costs were corresponding to the use of the type of database and the number of dynos, which is based on the usage demand of the game instances. There were over 1,000 eligible registered players in the UHM instances, so higher provision was needed. Since EWC and HNS instances only has 130 users and 10 users respectively, a lower provision was sufficient.

Before hosting the Makahiki instances on Heroku, we had to decide what are the appropriate provisions for these instances with acceptable run time performance. We did several performance tests with the Makahiki instance deployed on Heroku, and decided that 5 dynos was sufficient for servicing about 1,000 users in the first cloud instance, 2012 UHM Kukui Cup. We did not perform

extensive performance testing, knowing that we can always adjust the allocation of dynos during the competition if we observe the dynos are over-loaded or under-utilized. The performance tests were performed using the Apache JMeter [?]. [Figure 6.9](#) shows the results of running the tests with 2 dynos against 100 concurrent users and 5 dynos against 250 concurrent users.

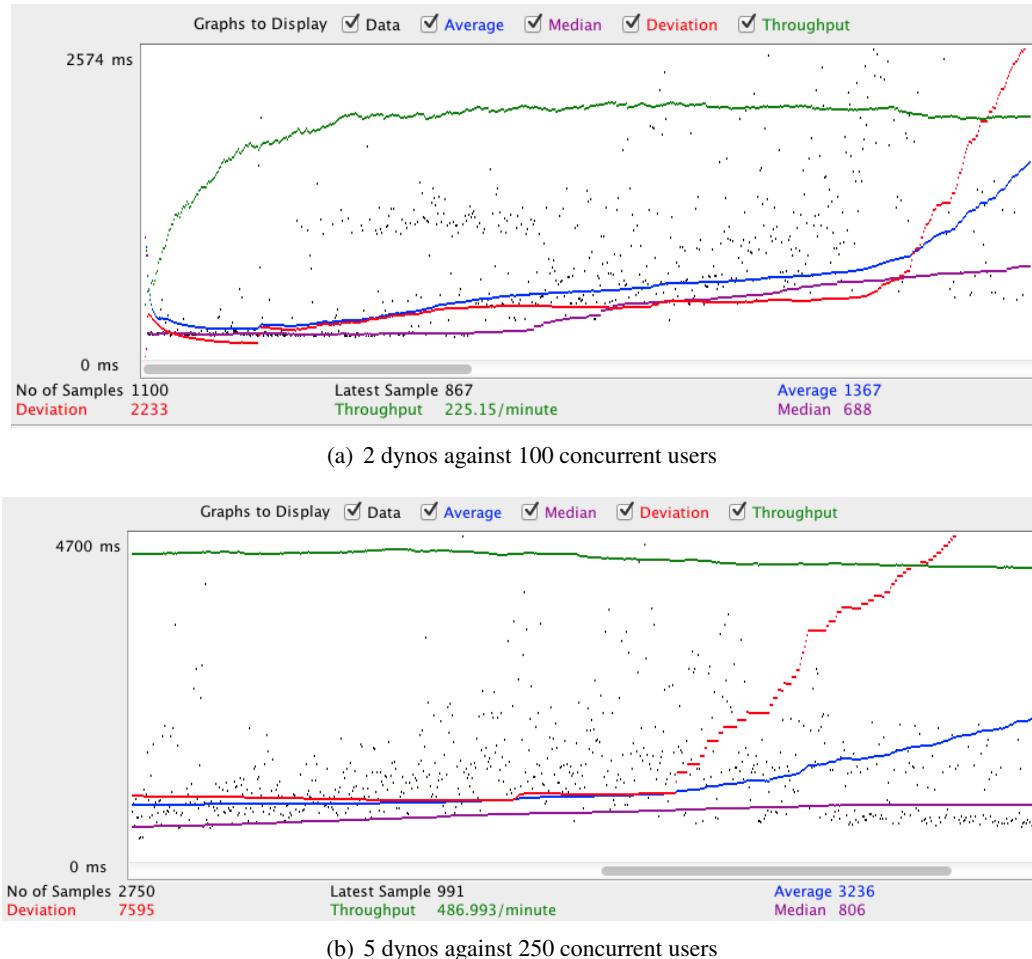


Figure 6.9: Performance testing to determine the appropriate number of dynos

We configured the Heroku cloud-based 2012 UHM instance to run with 5 dynos initially. After 5 days, we noticed that the dynos were under utilized, so we decreased the number of dynos to 4 and kept it run for about 1 month through the first 2 rounds. During the third round, the number of active players decreased, so we scaled down the Heroku dyno further to 2 dynos. At the beginning of the fourth round, we anticipated there may be a surge in website usage because of the new contents available in a new round, we scaled up the dynos to 4 again and found that after a while, the usage

was down thus we scaled down the dynos to 2 and kept the game running to the end. [Figure 6.10](#) shows the dynamic dyno allocation during the UHM 2012 instance

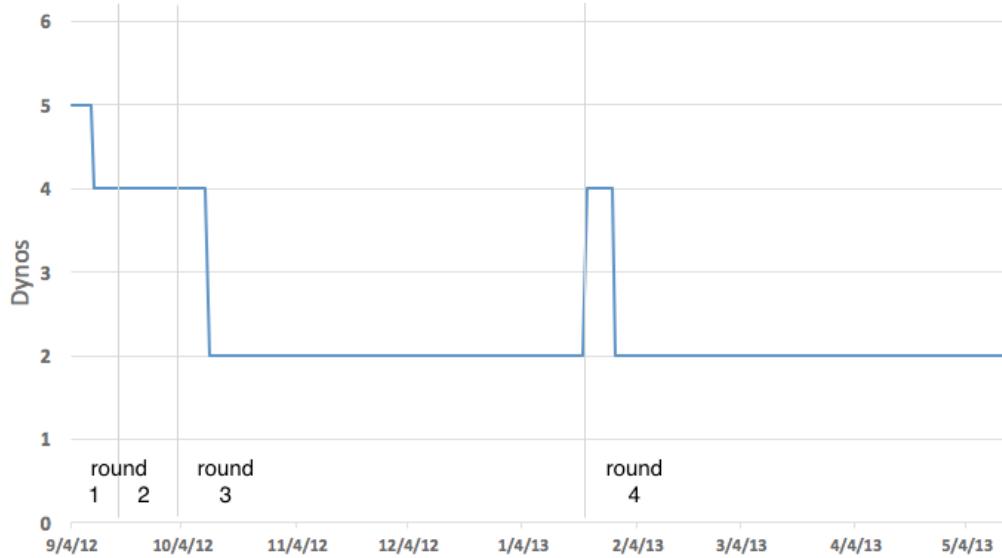


Figure 6.10: Dynamic Heroku dyno allocation during the UHM 2012 instance

In the 2014 UHM instance, which was also deployed on the Heroku cloud infrastructure for the similar size of user base as in the 2012 Kukui Cup, 4 dynos were allocated for the entire duration of the 2 weeks competition because of its short term nature. We also performed a experiment to see if there is any performance difference when using the upgraded “Standard 2” cloud database. We did not observe any noticeable performance improvement with the upgraded but much more expensive database. We consider the “Standard 0” database would be sufficient to support the 2014 instance, thus bringing the cost for this kind of Makahiki instance to around \$100 per month.

In the case of EWC2012 and HNS2013 instances, because the user base is much smaller, about 130 and 10 users respectively, we used a less expensive “Basic” Heroku cloud database and allocated 1 dyno most of the time to achieve acceptable application performance. The two weeks EWC2012 competition found that the 1 dyno provisioned was sufficient without any performance complaints. The HNS2013 instance was initially provisioned with 1 dyno. After three months, the game manager encountered occasional slow loading of the smart grid game page, so we increased the number of dynos to 2. Later on, we noticed the usage of the game site was low so we decreased

the number of dynos back to 1 without affecting the running game. On average, the HNS instances use 1.3 dynos during the competition period.

#### 6.1.3.2 Amazon S3 Cloud Service

In addition to Heroku, Makahiki also integrates with other cloud services such as Amazon S3 [?]. During the deployment, Makahiki stores about 9 MB of static media files on Amazon S3. During the game, Makahiki uses S3 to store all the user uploaded medias such as player profile images, submission files, etc. Depending on the length of the game, number of players, and the number of submission files, the usage of S3 storage might be different between various instances. [Table 6.6](#) shows the Amazon S3 storage usage for the 4 real world Makahiki instances. According to S3 pricing [?], the charge for the first 1 TB per month is \$0.03 per GB, and \$0.004 per 10,000 GET requests. With this very low cost of using S3, the cost of using S3 for real world Makahiki instances was negligible.

Instances	Amazon S3 storage	Cost
UHM2012	204M	\$0.01
UHM2014	101M	\$0.008
EWC2012	11MB	\$0.004
HNS2013	10MB	\$0.004

Table 6.6: Amazon S3 storage usage for Makahiki instances

#### 6.1.3.3 Memcache Cloud Service

Another cloud service that Makahiki used is the memcache service. Several memcache services are provided via Heroku add-ons to applications hosted on Heroku. By default, Makahiki used the free tier memcache add-on from one of the service providers called MemCachier[?].

MemCachier's free tier memory cache service is limited to 25MB. Depending on the size of memory cache used by the Makahiki instance, the organization could choose to use a different tier of cache service. In Makahiki, memory cache is mainly used to store runtime system-wide information and game data from online active players. The data in the cache has a defined TTL (time to live) value so if the data is not being accessed after the TTL, it will be removed from the

cache. In the cases of the 4 real world Makahiki instances, 25MB is enough for running the game.

MemCachier provides a simple usage page to monitor the cache usage for your application, as shown in [Figure 6.11](#). We used this tool to identify the cache usage need for the Makahiki instance and monitor a running instance's usage.

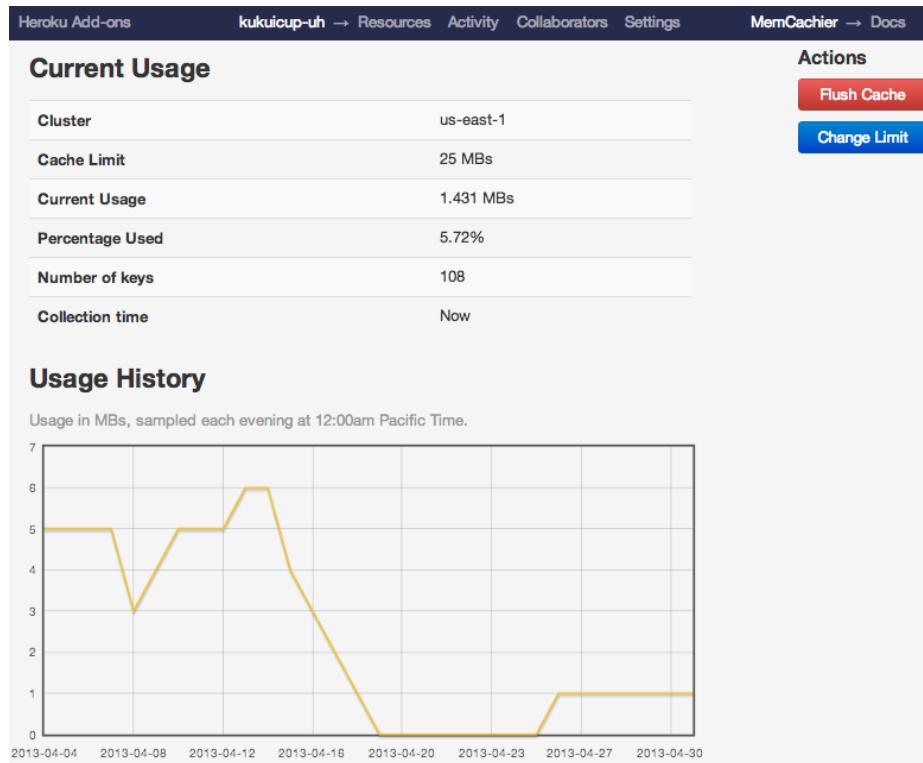


Figure 6.11: Memory cache usage monitoring

#### 6.1.3.4 Cloud Monitoring Service

Makahiki uses a cloud application monitoring service called NewRelic to help monitor the game instances that are deployed to Heroku. The monitoring instrument is enabled by using the NewRelic client to start the Makahiki application. Once monitoring is enabled, the NewRelic service monitoring console shows runtime metrics of the Makahiki instance. [Figure 6.12](#) and [Figure 6.13](#) shows the monitoring screens for the application server metrics and database metrics respectively. Using this monitoring service, we can identify the performance issues from various sources of a running instance. It was also used to monitor the utilization of the allocated resources.

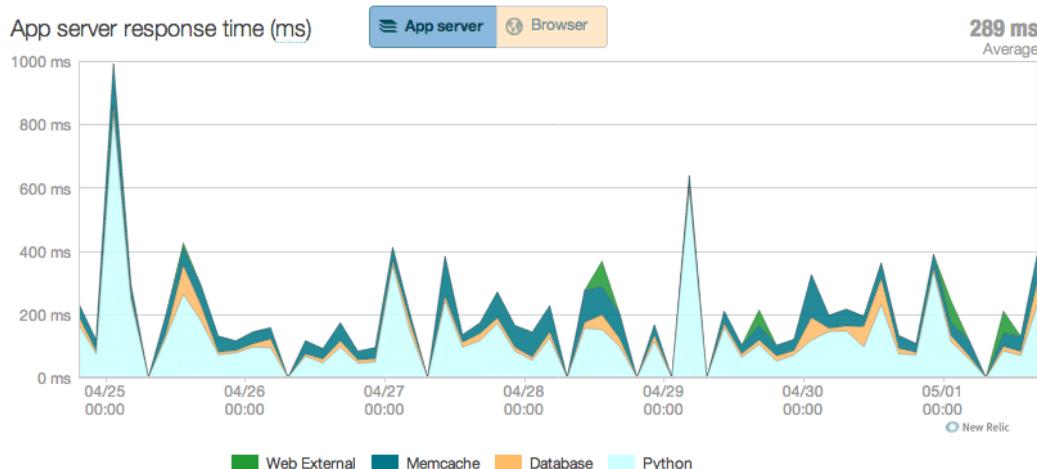


Figure 6.12: NewRelic application monitoring

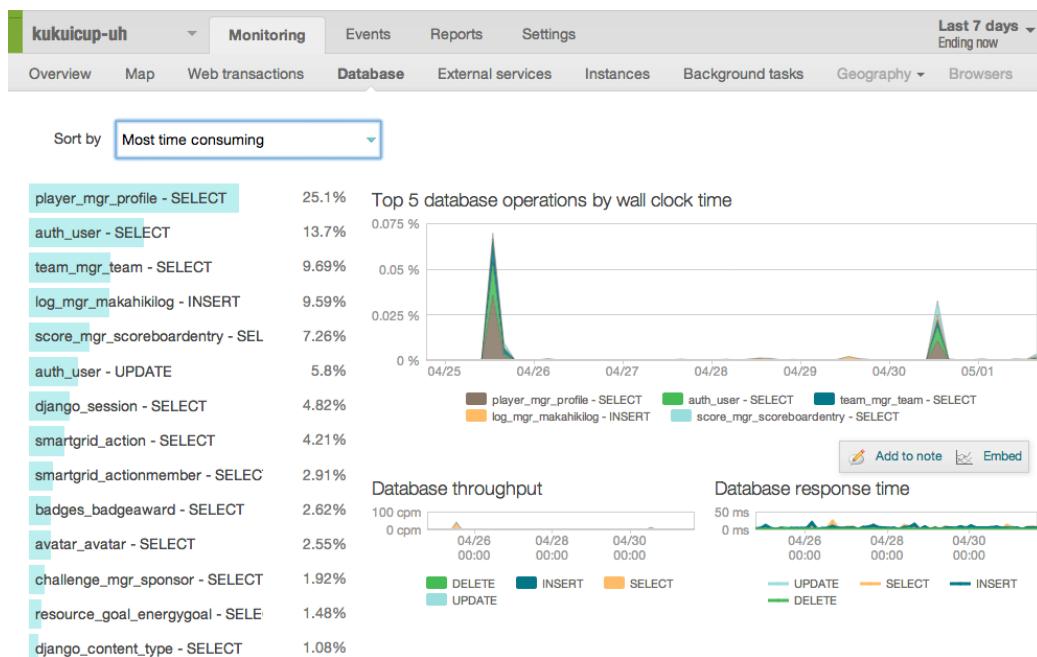


Figure 6.13: NewRelic database monitoring

In summary, this section described our experiences hosting the real world Makahiki instances in the Heroku cloud infrastructure. We compared the different resource provisions and associated costs for the different instances based on the number of players and the duration of the game. Even during the game, the dynamic resource allocation capability of the cloud infrastructure provided a flexible and better utilization for this type of serious game. The areas we need to consider for the

cloud infrastructure provisioning are:

- How many dynos / computing units to start with (for horizontal scaling)?
- What kind of dynos, 1X or 2X (for vertical scaling)?
- How big / powerful / reliable should the database be?
- How much memory cache do we need?
- What are the other cloud service (e.g, S3, monitoring) utilization?

#### **6.1.4 Summary**

This section described our experiences supporting seven real world Makahiki instances in four different organizations. The successful creation of these serious game instances provides evidence that Makahiki can be successfully tailored to the needs of different organizations. Among these instances, UHM and HPU used different metering infrastructure for energy data collection, EWC collected their resource data manually, and HNS did not have energy data at all. While UHM and HPU challenges involved only energy consumption data, the EWC challenge involved both energy and water consumption data. The IT infrastructure at UHM and HPU provided authentication services using CAS and LDAP, while EWC used both CAS and the built-in Django internal authentication, and HNS used only the internal authentication. Moreover, the user interface was customized to “brand” each challenge with the logo, thematic elements, and the education content of the sponsoring organizations.

Lastly, the hosting infrastructure among these instances are different in two categories: local hosting for HPU and the 2011 UHM instances, and cloud hosting for EWC, HNS and 2012, 2014 UHM instances. These two types of hosting requirements created unique challenges for Makahiki. There are different costs associated with these hosting solutions. Although there are some incurred costs in the case of hosting the Makahiki games in the Heroku cloud infrastructure, there is no need for maintaining local infrastructure, such as keeping the server running, upgrades, patching as well as locating physical space for the server. There is no need to allocate IT support staff resource to

do system admin. In addition, the dynamic resource allocation capability of the cloud infrastructure provides a flexible and efficient utilization of infrastructure resources in running the competition, especially in the case of short competition duration and varying user demands during the game.

## 6.2 SGSEAM Assessment Results

This section describes the result of applying the formal assessment method of SGSEAM to the Makahiki framework to assess the strengths and weaknesses of the Makahiki as a serious game framework.

The SGSEAM assessment plan for Makahiki ([Appendix B](#)) describes two categories of assessment approaches. The *in-vivo* approaches of pre-post effectiveness study, in-game surveys, and post-hoc interviews were applied to the real-world UHM, HPU and EWC Makahiki instances. The other *in-vitro* approaches were applied to the UHM ICS691 serious game course as an in-lab experiment. [Table 6.7](#) provides the overview of assessment and participants for each SGSEAM stakeholders of the Makahiki framework.

Stakeholder	Assessment Approach	Participants
Players	Pre-Post effectiveness study	UHM Kukui Cup players
	Self-reported effectiveness survey	
	Self-reported usability survey	
	Engagement metrics	
System admins	In-lab installation study	UHM ICS691 students
	Post-hoc system admin interview	HPU Kukui Cup sysadmin
Game designers	In-lab game design study	ICS691 students
	Post-hoc game designer interview	HPU & EWC Kukui Cup game designers
Game managers	Post-hoc game manager interview	HPU & EWC Kukui Cup game managers
Developers	In-lab game development study	ICS691 students

Table 6.7: SGSEAM assessments for Makahiki

As the assessment approaches were carried out, the assessment data were collected and stored in the data repository ([Appendix C](#)). The following sections describe the data analysis and the results for the assessment of each stakeholder.

### **6.2.1 Makahiki Player Assessment**

We used four approaches to assess the player experience for the Makahiki framework. They are a pre-post effectiveness study, a self-reported effectiveness survey, a self-report usability survey, and engagement metrics. The real world Makahiki instances of the UHM Kukui Cup challenge were used for the Makahiki player assessments.

#### **6.2.1.1 Pre Post effectiveness study**

In the 2011 Kukui Cup Challenge at the University of Hawaii at Manoa, there were over 1000 eligible players, who were mostly first year college students living in the 5 residence halls. The challenge was designed to last for 3 weeks and the student players were divided into 20 teams based on the dorm locations where they resided. Each team's energy consumption was measured by a smart meter installed in various locations inside the resident hall. Makahiki recorded the energy consumption for the players before, during and after the challenge. Makahiki also recorded detailed logging data from every interaction between the players and the website.

To assess the effectiveness of the framework for designing games that improve player literacy in sustainability, two energy literacy surveys were conducted, one before the challenge (pre-game) and one after the challenge (post-game).

Robert Brewer designed and conducted the survey. The results are reported in his dissertation [?]. 24 players completed both surveys. Out of the total 19 energy literacy questions, the average number of questions answered correctly is 7.54 before the challenge, and 8.96 after the challenge. This result indicates an 18% ( $p=0.056$ ) improvement on the energy literacy. Non-players as a control condition were also surveyed and found that their literacy did not change. The result is shown in the [Figure 6.14](#). According to Brewer [?], “Based on the questionnaire results, it appears that the energy knowledge of challenge participants increased modestly compared to those that did not participate in the challenge.”

To assess the effectiveness of the framework for designing games that reduce players' resource consumption, The energy usage data that was collected before, during and after the challenge were used to compare the differences. Before the challenge, an energy usage baseline was established.

Group	Pre-challenge		Post-challenge		% Change
	Mean	Std. Deviation	Mean	Std. Deviation	
Non-challenge participants	7.46	2.377	7.37	2.570	-1.2%
Challenge participants	7.54	1.837	8.96	3.290	18.8%

Figure 6.14: Literacy survey result from Brewer's dissertation [?]

Robert Brewer calculated the energy consumption data before and after the UHM 2011 Kukui Cup challenge. According to his dissertation [?], 12 out of the total 20 teams reduced their energy consumption compared to the baseline. The highest reduction was 16.1%. However, 3 teams actually increased their energy consumption, with the highest increase of 11.7%. The result is shown in the [Figure 6.15](#). Overall, the average reduction of the 20 teams was low (1.9%). According to Hodge's North America collegiate energy competition survey [?], the average reduction in electricity use for these competitions is 9%. This study did not provide strong evidence that the challenge results in effective reduction of energy use.

Lounge	Conservation
Ilima-A	16.1%
Ilima-E	6.4%
Lehua-E	6.1%
Lehua-A	5.5%
Ilima-C	5.3%
Mokihana-D	2.5%
Lehua-C	2.4%
Ilima-D	2.3%
Mokihana-E	2.1%
Mokihana-C	1.5%
Ilima-B	1.3%
Lehua-D	0.5%
Mokihana-B	-3.4%
Mokihana-A	-8.4%
Lehua-B	-11.7%

Figure 6.15: Energy consumption result from Brewer's dissertation [?]

In summary, this study indicates that Makahiki was successful in achieving player literacy improvement. It did not provide strong evidence that the challenge results in effective reduction of energy consumption by players.

### 6.2.1.2 Self-reported effectiveness survey

A survey to gather the opinions of participants regarding the effect of the challenge to their sustainability awareness and behavior changes was included during the last round of the challenge. Two survey questions were used to assess the self-reported perception of the players regarding their interests in energy conservation and sustainability prior and after playing the Kukui Cup.

In the 2011 UHM Kukui Cup challenge, 43 players completed the survey. The responses to the survey were provided in free text. We analyzed the free text responses and coded them into three categories: “Yes”, “No”, and “Somewhat”. [Table 6.8](#) lists results of the responses. [Figure 6.16](#) illustrates the percentages of self-reported interests in the sustainability prior to the KuKui Cup and the effect after playing the Kukui Cup.

Question	Number of Responses		
	Yes	No	Somewhat
Prior to playing the Kukui Cup, were you interested in energy conservation?	24	8	11
Has the Kukui Cup increased your interest in energy conservation and sustainability?	37	0	6

Table 6.8: Interests in sustainability prior and after the KC (2011 UHM, n=43)

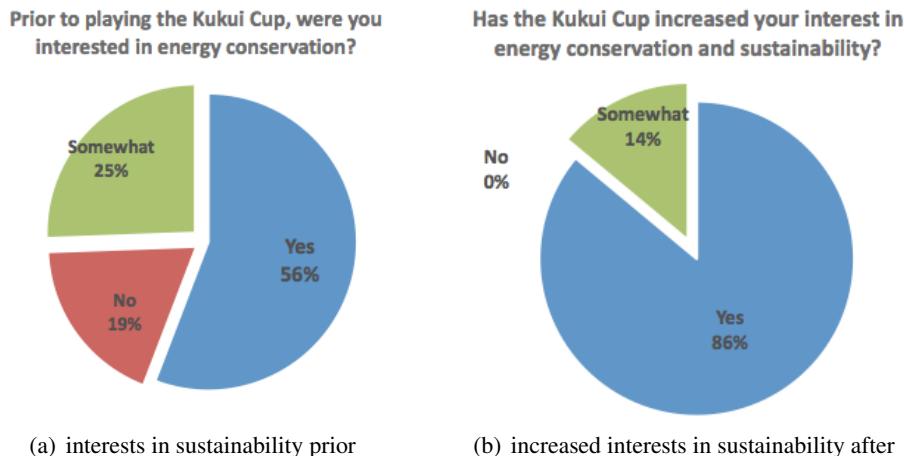


Figure 6.16: Interests in sustainability prior and after the UHM 2011 KC

The self-reported responses indicates that there are 19% players who were not interested in the sustainability prior to the Kukui Cup. After playing the Kukui Cup, 100% of responses reported an

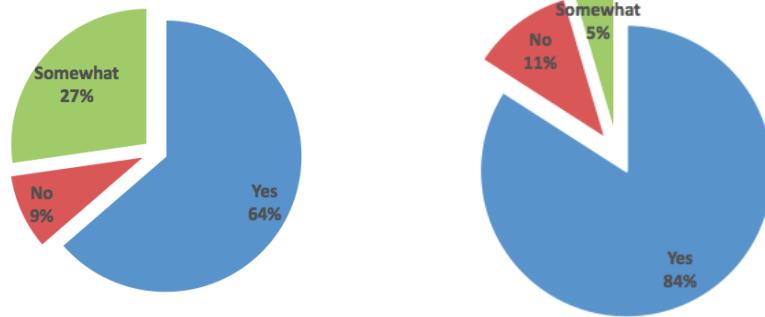
affirmative or somewhat affirmative increase of interests in sustainability.

In 2012 UHM Kukui Cup challenge, the similar survey questions were included the last round. 44 players completed both the two questions. **Table 6.9** lists results of the responses. **Figure 6.17** illustrates the percentages of self-reported interests.

Question	Number of Responses		
	Yes	No	Somewhat
Prior to playing the Kukui Cup, were you interested in energy conservation?	28	4	12
Has the Kukui Cup increased your interest in energy conservation and sustainability?	37	5	2

Table 6.9: Interests in sustainability prior and after the KC (2012 UHM, n=44)

Prior to playing the Kukui Cup, were you interested in energy conservation?      Has the Kukui Cup increased your interest in energy conservation and sustainability?



(a) interests in sustainability prior

(b) increased interests in sustainability after

Figure 6.17: Interests in sustainability prior and after the 2012 UHM KC

The self-reported responses from 2012 KC indicates that there are a smaller percentage (9%) of players than 2011 KC (19%) that were not interested in the sustainability prior to the Kukui Cup. This could be due to the increasing sustainability exposure over years to the young students. After playing the Kukui Cup, 89% of responses reported an affirmative or somewhat increase of interests in sustainability. There are 5 players (11%) reported no changes of interests in sustainability. A closer look at the data indicates that the 5 players that reported no changes of interests also answered “Yes” or “Somewhat” interested to the sustainability prior to the Kukui Cup. The 4 players that reported no interests in sustainability also reported that playing KC had increased their interests.

**Figure 6.18** shows some sample responses from the students who considered the Kukui Cup had increased their interests in sustainability:

1. “Yes. I’m more aware of everything I am using.”
2. “I’ve been interested, but the Kukui Cup has expanded and broadened my mind on what else I can be doing to help with conservation and sustainability. I think people are genuinely concerned, they just don’t know how to exactly conduct them, or what other ways they can do it.”
3. “Yes, it made me realize that you can save a lot of money by turning off and unplugging a few things. (Hopefully lower tuition? =D )”
4. “It has increased my interest since I didn’t know much about how much energy were using.”
5. “Some workshops made me think differently of how I use energy. Makes alternatives more interesting for me.”
6. “It taught me some new interesting facts about renewable energy and made me more aware.”

Figure 6.18: Sample responses of reported increased interests in sustainability

There was a question in the 2011 KC survey asking about the players’ perception about how would they describe the Kukui Cup. The players were asked to check all that applies to their agreement to the following descriptions of the Kukui Cup: “Educational”, “Fun”, “Addictive”, “So-so”, “Difficult”, “Boring”, “Not useful”, and “Other” where they will type in their free text responses. There were 43 responses from the 2011 KC survey. The numbers of responses and their percentages are listed in **Table 6.10** (sorted by percentage in descending order).

Question: How would you describe the Kukui Cup?	Number of Responses	Percentage
Educational	41	95%
Fun	39	91%
Addictive	19	44%
So-so	9	21%
Other	5	12%
Difficult	3	7%
Boring	1	2%
Not useful	0	0

Table 6.10: Self-reported Perception of the Kukui Cup in 2011 UHM KC (n=43)

The majority of the responses indicated the players perceived the Kukui Cup as “Educational” (95%) and “Fun” (91%). 44% of players perceived Kukui Cup as “Addictive”. On the other hand,

1 player (2%) considered the Kukui Cup as “Boring”. The “Other” responses are: “AWSOME-NESS”(1), “engaging”(1), “fun competition”(1), “Great way to bond with others”(1), “impressive”(1).

Another survey question was asked about the players’ self reported behavior change during the challenge in the 2012 UHM Kukui Cup. The question is “Did you change your behavior during the competition based on the commitment(s) you made? if so, how?” It is a free response question. 45 players completed the survey. The free text responses were categorized into three categories: “Yes”, ‘already a habit’, and “no”. The results are listed in [Table 6.11](#).

<b>Question: Did you change your behavior during the competition based on the commitment(s) you made?</b>	<b>Number of Responses</b>	<b>Percentage</b>
Yes	39	87%
Already a habit	4	9%
No	2	4%

Table 6.11: Self-reported Behavior Changes in 2012 UHM KC (n=45)

The majority of the responses indicate some kind of behavior changes during the competition. [Figure 6.19](#) shows samples of the response that considered behavior changes had happened during and after the competition.

- 1. “I changed my behavior during the commitments but found myself doing the same things i did before after they were over.”
- 2. “Yes, because knowing the facts of how much energy we use has helped me realize that I have been wasting money and energy.”
- 3. “yes, my behavior has changed during the competition. I’m more aware of the things i do such as turning off appliances when not in use, as well as using more natural energy such as sunlight, rather than electricity.”

Figure 6.19: Sample responses of reported behavior changes

In summary, the self-reported effectiveness survey results indicate that the players of Makahiki considered their experiences to be positive and there were impacts to their sustainability awareness and behaviors because of playing the Makahiki games.

### 6.2.1.3 Self-reported usability survey

A survey to gather the opinions of players regarding the usability of the website was added during the last round of the challenge. The players were asked to rate how much you agree with the following 4 usability statements in a Likert scale (“Strongly disagree”, “Disagree”, “Neutral”, “Agree”, “Strongly agree”):

1. It was easy to find what I was looking for in the website.
2. The website was responsive. I did not wait too long after I clicked on something.
3. The website provided adequate help in teaching me how to play the game.
4. I understood the rules of the game and how to play.

The questions were asked in both the 2011 and 2014 Kukui Cup at the University of Hawaii at Manoa. 43 players completed the survey in the 2011 KC while 18 players completed it in the 2014 KC.

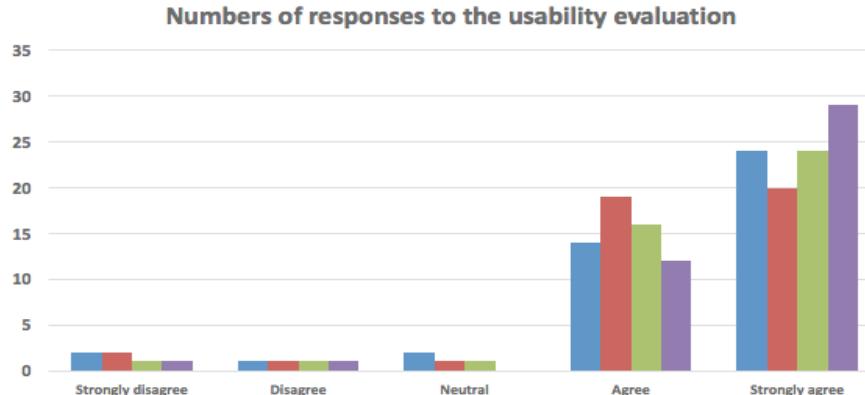
[Table 6.12](#) and [Table 6.13](#) shows the results of the self-reported usability responses for 2011 and 2014 KC respectively. [Figure 6.20](#) illustrates these responses in a graphical format.

Usability statement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
It was easy to find what I was looking for	2	1	2	14	24
The website was responsive	2	1	1	19	20
The website provided adequate help	1	1	1	16	24
I understood the rules of the game	1	1	0	12	29

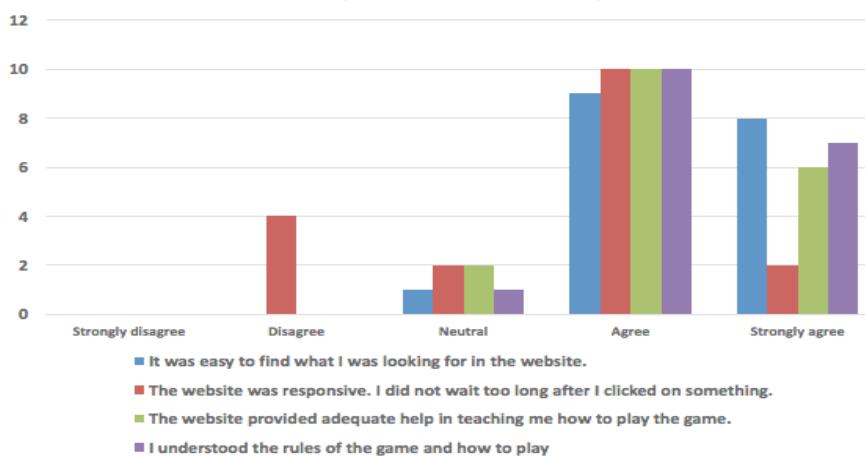
Table 6.12: Self-reported Usability in 2011 UHM KC (n=43)

Usability statement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
It was easy to find what I was looking for	0	0	1	9	8
The website was responsive	0	4	2	10	2
The website provided adequate help	0	0	2	10	6
I understood the rules of the game	0	0	1	10	7

Table 6.13: Self-reported Usability in 2014 UHM KC (n=18)



(a) 2011 Self-reported Usability Measurements(n=43)



(b) 2014 Self-reported Usability Measurements(n=18)

Figure 6.20: Self-reported Usability Measurements in 2011 KC and 2014 KC

The majority of the players reported that they agreed the usability of the website is good with one exception of responsiveness from the 2014 responses. 4 out of 18 players (22%) of the 2014 KC reported that they considered the website is not very responsive. Compared to the result of the 2011 KC, this indicates a perceived performance downgrade in the 2014 KC website.

Another question was asked in the 2014 UHM KC survey regarding the issues encountered when using the website. 3 out of 18 responses (17%) reported that the loading of the website page was slow. This confirms the responsiveness issues discovered in the survey question described above. Another issue reported is confusion regarding the scoreboard's display on the website. 3 out of 18 responses (17%) reported that scoreboard's display kept changing so it is not easy to see rankings. 2

responses reported that some of the videos were not displayed. It was later discovered that the links to those videos were outdated.

In summary, the self-reported usability survey is a good tool to get feedback from the players of Makahiki regarding the usability of the game. It reveals that most of the usability in Makahiki is good. It discovered a few usability issues in 2014 KC such as the slow performance of certain pages, the confusion of scoreboard display, and the broken links to the educational video.

#### **6.2.1.4 Engagement metrics**

Based on the engagement metrics proposed in SGSEAM (4.3.2.1.3), we calculated a variety of engagement metrics to assess the player's engagement level in the 2011 and 2012 UHM Kukui Cup challenges created by the Makahiki framework. The metrics are calculated by analyzing the website logs and the game data collected by the application. The results are shown in Table 6.14.

<b>Measurement</b>	<b>2011 KC</b>			<b>2012 KC</b>		
	<b>MIN</b>	<b>AVG</b>	<b>MAX</b>	<b>MIN</b>	<b>AVG</b>	<b>MAX</b>
Participation rate	13%	37%	74%	19%	34%	64%
Number of players per day	43	85	147	0	12	130
Play time per day	1 min	27.7 mins	8.5 hours	0	6.2 mins	8.8 hours
Submissions per day	32	266	1110	0	30	953
Social interactions per day	51	208	468	0	31	502
Website errors per day	0	0.6	4	0	2	458

Table 6.14: Engagement Metrics for 2011 and 2012 UHM Kukui Cup

The participation rate is the percentage of players who played the game. The 2011 UHM KC had a 37% average participation rate, while the 2012 KC had 34%. According to the report from the successful Oberlin College Energy competition [?], 46% of residents looked at the competition website at least once (based on the IP addresses of the web server requests). Because the Oberlin system did not support user accounts, the IP address may not be a very accurate method to identify unique players. The actual participation rate might be lower than 46%. Compared to the average 11% participation rate of Western Washington University's Go for Green challenge [?] and the 0.4% participation rate of University of Southern California's Ecolympics challenge [?], both the 2011 and 2012 UHM KC challenges had good participation.

Over the 3 week challenge in the 2011 KC, an average player spent about 27.7 minutes per day on the website, while in the 24 week challenge in the 2012 KC, an average player spent about 6.2 minutes per day. Both instances had one player spending over 8 hours (8.5 and 8.8 respectively) on one day, which is quite significant amount of time for a student to spend in this kind of serious game. The daily minimum time spent on the 2011 KC is 1 minute which indicates that for every day during the challenge, there was at least one player who played at least 1 minute. The daily minimum time spent on the 2012 KC is 0 minutes. This indicates that there was at least one day when no player used the website.

In order to investigate engagement in more detail, I looked at engagement measurement in a time series format. [Figure 6.21](#) shows the timed measurements graph during the 2011 and 2012 Kukui Cup.

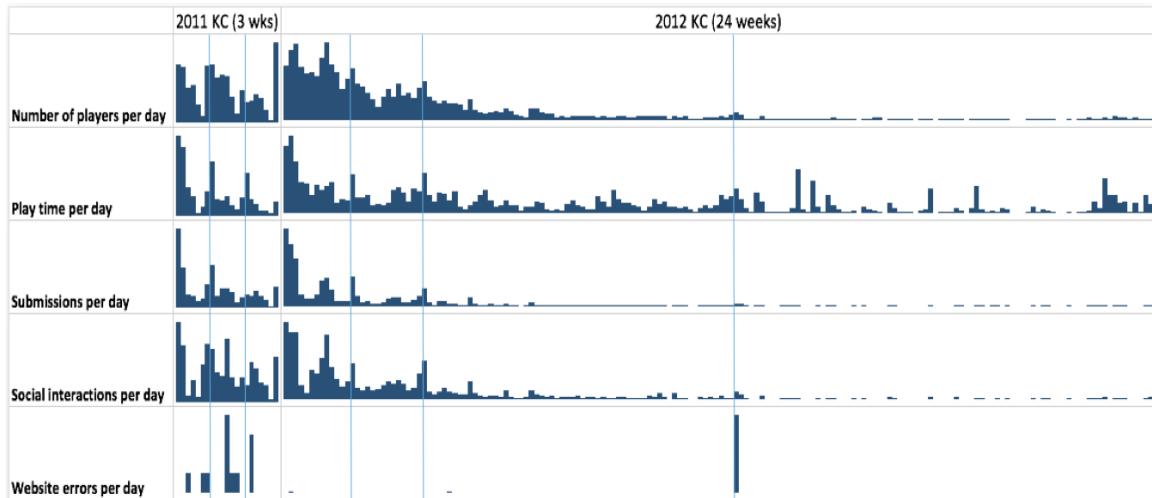


Figure 6.21: Engagement Measurements during the 2011 and 2012 UHM Kukui Cup

The vertical lines in the graphs indicates the division of the rounds for the 2011 and 2012 KC. In the 2011 KC, there are 3 rounds, each round lasting 1 week. In the 2012 KC, there were 4 rounds, the first and second round lasted 2 weeks each, and the third round lasted about 8 weeks, while the last round lasted about 12 weeks, for a total of 24 weeks for the 2012 KC.

In addition to the daily number of players and daily play time which indicate the level of visits to the website or game, the daily submission and daily social interaction measurements indicate the level of deeper interactions with the game. We can see that during the 3 short weeks of the 2011

KC, the engagement patterns are similar in each week. The first few days have higher levels of interactions then decrease later into the week. There is a spike on the last day of the round, which may be due to last minute surge of interest in winning the round.

We can also see that in the 2012 KC, the engagement level significantly dropped after the second round, which is 4 weeks after the beginning of the game. There are still some players spending time on the game but the number of submissions and social interactions had decreased significantly. The number of players decreased in the third round and decreased even more in the last round. It is interesting to see that there are still some amount of play time during the third and fourth round. A closer look at the data indicated that they are time spent from the few top players who have a chance to win the game.

Although the website errors in the 2012 KC shown in [Table 6.14](#) seems high, the detailed graph in [Figure 6.21](#) shows that the errors mostly happened during one day which is the day after the start of the last round. The investigation of the error in the log file revealed that they are due to a content configuration error in a newly available event in the last round. The error caused the players to not be able to submit the completion of the event to claim their points so they kept trying and encountered the error repeatedly. The error was corrected 2 hours later after the first such error occurred.

In summary, this analysis indicates that Makahiki was successful in achieving player engagement for a 3-to-4 week period in this kind of serious game. After this period, engagement seems to taper off. It is possible that engagement might stay high for a long period if there are a lot of activities throughout the period, but it requires substantial effort from game designers to create new activities and game managers to manage the game. More research must be done to understand how to maintain high levels of engagements over a long term period.

### **6.2.2 Makahiki System Admin Assessment**

We used two approaches to assess the system administrators' experience with the Makahiki framework. They are the in-lab installation study and the post-hoc system admin interview.

### 6.2.2.1 In-lab installation study

In the in-lab installation study, the participants are the students in a serious game class (ICS691) in Spring 2013 in the information and computer sciences department at the University of Hawaii at Manoa. The students were tasked with installing the Makahiki system into their local computers as well as deploying to the Heroku cloud environment. A Google Form (described in [Section G.1](#)) was used to ask the students to record the time they spent completing each step and the problems they encountered. The students also provided feedback about their installation experiences in the form of blog posts. There were a total of 8 students who voluntarily participated in the experiments. The participants were either senior undergraduates or graduate students majoring in Computer Science.

In the local installation study, the results from the Google Form responses show that the average total time to successfully install Makahiki in their local computers was 1.4 hours, with a maximum time of 2 hours and a minimum time of 0.9 hour. [Table 6.15](#) and [Figure 6.22](#) shows the average time and their percentage distributions of each step in the local installation study.

Category	Installation Tasks	Average Time (minutes)
Setup runtime environment	Install Python, C compiler, Git, Pip and Virtual environment	10.0
Install dependencies	Install Python Imaging Library, Memcache, required python packages	29.4
Install and configure database	Install and configure PostgreSQL	37.5
Download the software	git clone from Github	7.5
Install the software	setup environment variables, run initialize_instance script	26.8
Start the server and verify	start the server and verify the server running correctly	2.5

Table 6.15: Average time (minutes) for local installation steps (n=8)

The descriptive problems reported by the students in both the Google Form and their blog posts were coded and categorized. The analysis results are shown in [Table 6.16](#).

From the above analysis, we identified that the “Install and configure database” step of the local installation has the longest average time. It also has the most problems reported from the participants. This assessment determines the areas for future improvement for the local installation

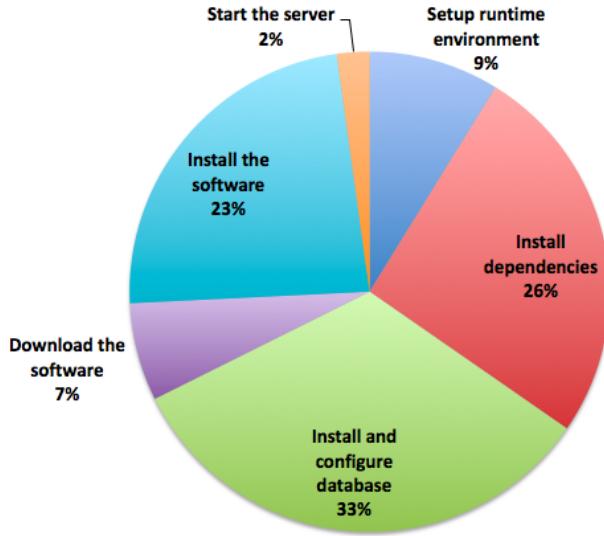


Figure 6.22: Average time for local installation steps (n=8)

Problem encountered	Number of participants
Cannot find configuration file to edit during database installation	4
Documentation of install script is confusing about creation of the DB user	2
More parts of installation could be covered by install script	2

Table 6.16: Makahiki local installation issues reported (n=8)

are (1) to improve documentation on DB installation, and (2) to improve the install script to automate more installation tasks.

In the Heroku cloud installation study, the results from the Google Form responses revealed different experiences. The average total time to successfully install Makahiki in the Heroku cloud environment was 1.9 hours, with a maximum time of 3.6 hours and the minimum time of 0.7 hour.

**Table 6.17** and **Figure 6.23** shows the average time and their percentage distributions of each step in the Heroku cloud installation study.

**Table 6.18** shows the result of the analysis from the blog post feedbacks regarding participants' cloud installation experiences.

From the above analysis, we identified that the "Setup cloud environment" step has the longest average time. It also has the most participant-reported problems regarding confusion in setting up Amazon S3. This assessment reveals the areas for future improvement for the cloud installation are:

Category	Installation Tasks	Average Time (minutes)
Setup cloud environment	Install Heroku client, add ssh keys to Heroku, verify Heroku account, setup Amazon S3	50.6
Download software	git clone from Github	6.3
Deploy to cloud	setup environment variables, run initialize_instance script	39.5
Start the server and verify	start the server and verify server running correctly	29.4

Table 6.17: Average time (minutes) for Cloud installation steps (n=8)

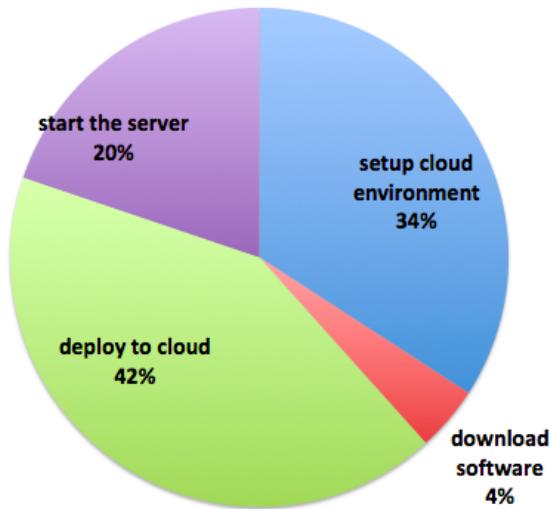


Figure 6.23: Average time for cloud installation steps (n=8)

Problem encountered	Number of participants
confusion in setting up Amazon S3	5
took a long time to upload application to Heroku	3
needing to use the virtual environment is not documented	3
Failure messages are buried in the verbose script output thus overlooked	2

Table 6.18: Makahiki cloud installation issues reported from blog posts (n=8)

(1) to improve documentation on cloud environment setup; (2) to improve usability of the install script; and (3) develop tools to verify and diagnose the environment setup.

In summary, the in-lab installation study identified database installation, documentation and the

usability of the install script as the weak points in the process of installing Makahiki. Otherwise, SGSEAM indicates generally positive results regarding Makahiki installation.

### **6.2.2.2 Post-hoc System Admin Interview**

In order to gain insight into the experience of real world system admins who use Makahiki, I performed interviews with the system admin of the Hawaii Pacific University (HPU) Kukui Cup challenges and analyzed the problems they encountered during software installation and system administration. The HPU KC server software was installed by the system administrator from the HPU IT department with assistance from the Makahiki development team. The installation was performed in the HPU local IT infrastructure and included integration with the HPU LDAP server and their email server. During the installation, issues encountered were communicated through emails and phone calls between the administrator and Makahiki development team. The interview took place after the challenge.

The data about the system admin experience was compiled and listed in [Table 6.19](#) and [Table 6.20](#) for the 2012 and 2013 HPU KC instances respectively. The working days to complete is the working days for the HPU system admin to complete an installation task. During those working days, the system admin might have other work responsibilities not related to the Makahiki software installation. The actual time spent working on the Makahiki installation might be less than the reported working days, but this data may reflect the actual work load in a typical IT organization, so it is still an indication of relative time spent on the different installation tasks.

The data shows that the the configuration to use HPU's LDAP server and email server are the most difficult tasks. It took a comparable large amount of time to install the LDAP libraries, create the ldap account, test the correct configuration and eventually discover that the Makahiki software needed to be modified to support HPU's special server settings.

Once the integration with the HPU's local infrastructure was completed in 2012, the 2013 experience was much easier, and the LDAP and email server configuration changes were completed quickly.

When asked about the experience of maintaining the Makahiki server, such as backup and mon-

<b>Installation Task</b>	<b>Time to Complete</b>	<b>Issues and Problem Encountered</b>
Install and startup server	5 days	error in installing the runtime environment virtualenvwrapper
Install image library	1 day	error in displaying JPG image during testing
Integrate with HPU LDAP server	20 days	need to install ldap libraries; need to create a special bind user to connect to the ldap server; use the correct ldap DN; HPU ldap server use non-standard way to identify user so the Makahiki software need to update to support HPU's server settings
Integrate with HPU Email Server	2 days	need to create a special email account for KC admin; HPU email server reported error due to the "from" parameter is not the same as the email account so the Makahiki software need to fix to support HPU's email.
Update the Makahiki software	1 day	none
Use SSL	not complete	need to procure the SSL certificate; the SSL certificate acquired by HPU does not have a trusted CA, it is decided that the competition will start without SSL
Total	29 days	

Table 6.19: Installation Issues in HPU 2012 KC

<b>Installation Task</b>	<b>Time to Complete</b>	<b>Problem Encountered</b>
Startup from last year's Vmware image	1 day	none
Makahiki software upgrade	1 day	need to recreate database migration script due to HPU still use older Postgresql DB version.
Integrate with HPU LDAP server	1 day	not able to login because the HPU LDAP server changed to use a new directory structure
Integrate with the new LDAP server	1 day	need to reconfigure to use the new LDAP server
Use HPU Email Server	2 days	not able to send email because the HPU email server configuration change.
Total	6 days	

Table 6.20: Installation Issues in HPU 2013 KC

itoring, the system admin answered that it was fairly straightforward to perform the backup. Since the server was built on the VMWare virtual machine, he used the snapshot function in the VMWare management console to perform the daily backup. The backed up image was successfully used in

2013 as the starting point for the 2013 KC instance without the need to re-install the software. The system admin did not do any monitoring and relied on the game designer and manager to report any issues to him. During the running period of the two challenges, there was no report of performance issues. Only one issue related to system installation was reported by the game designer during the testing of the 2012 HPU KC. It is the JPG image support library not being installed. It took 1 day for the sys admin to correct the problem.

In summary, the SGSEAM post-hoc system admin interview approach revealed difficulty in integrating Makahiki with an organization's LDAP and email server as well as using SSL in a local installation scenario. As a result, it is advisable to plan well ahead and allow ample time (at least one month) for these tasks. Compared to the in-lab installation study, the database installation is not an issue for the real world system admin. It is probably due to the IT system admin skill and experience difference between IT professionals and graduate students.

### **6.2.3 Makahiki Game Designer Assessment**

#### **6.2.3.1 In-lab game design study**

The in-lab game design study was also performed using the UHM ICS691 course in Spring 2013. One of the class assignments for the students in the experiment was to design a serious game using the Makahiki framework. We asked the students to follow specific design steps and record the time required and any problems encountered during their design process, using a Google Form outlined in [Section G.2](#). In addition, students were asked to provide feedbacks about their design experiences in the form of blog posts.

The game designer assessment was generalized into 7 tasks corresponding to distinct types of administrative tasks and game design planning. The time for each task is calculated from the Google Form responses.

There were a total of 8 students who voluntarily participated in the experiments. The most time consuming task was “Smart Grid Game Design”, which took an average of 107.9 minutes (56% of total time) to complete, while the least time consuming tasks was “Raffle Game Design”, which took an average of 7.9 minutes (7% of total time) to complete. [Table 6.21](#) and [Figure 6.24](#) shows

the average time and their percentage distributions for each design tasks.

Category	Design Tasks	Average Time (minutes)
Configure challenge settings	customize challenge name, logo; define teams, setup users, select games	46.5
Design the resource goal game	customize the default energy goal game with manual data input	5
Design the smart grid game	add a new level to the existing smart grid game with multiple types of actions and unlocking sequence	107.9
Design the top score game	add new prizes to the existing top score game	10.7
Design the raffle game	add new raffles to the existing raffle game	7.9
Design the badge game mechanic	add new badges with awarding condition	13.6

Table 6.21: Average time (minutes) for design tasks (n=8)

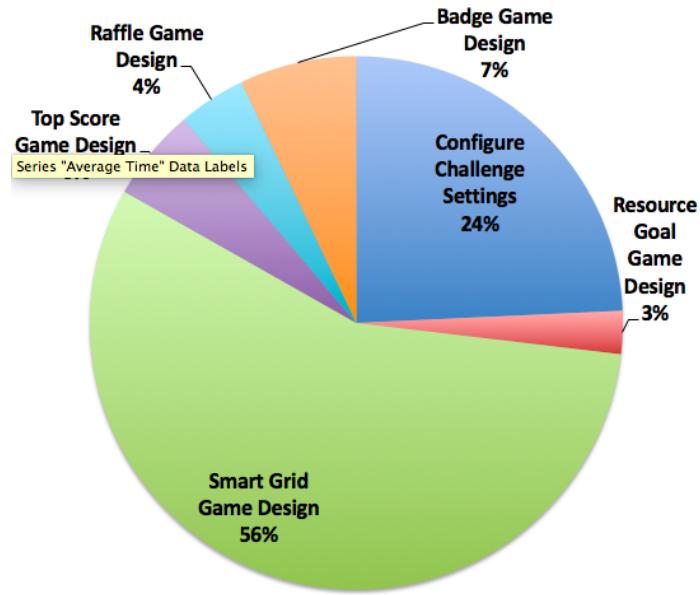


Figure 6.24: Average time for design tasks (n=8)

Problems reported in the blog post feedbacks were analyzed. The results are shown in Table 6.22.

In summary, the SGSEAM in-lab game design study revealed the two most time consuming tasks in Makahiki design are “Smart Grid Game Design” and “Configure Challenge Settings”. Is-

<b>Problem encountered</b>	<b>Number of participants</b>
Difficulty in understanding predicate system and unlock condition	7
A bug that prevented users with usernames containing capital letters from logging in	2
A bug in the processing of Ajax queries	1
Difficulty in generating event attendance codes for game activities	1

Table 6.22: Makahiki Game Design Analysis, (n=8)

sues encountered in “Smart Grid Game Design” include: 1) difficulty and lack of documentation on the predicate system used to define dependencies between game activities, and 2) difficulty in generating event attendance codes for game activities. Issues encountered in “Configure Challenge Settings” include: 1) a bug in the processing of Ajax queries caused by consecutive clicks on the same interface button, and 2) a bug that prevented users with usernames containing capital letters from logging in.

#### 6.2.3.2 Post-hoc Game Designer Interview

In order to gain insight into the experience of a real world game designer who uses the Makahiki software to design a serious game, I performed interviews with the game designers of the two Kukui Cup challenges in Hawaii Pacific University (HPU) and East-West Center (EWC) in the year of 2012. The game designer for HPU is the sustainability coordinator for the Hawaii Pacific University; while the game designers for EWC are two sustainability coordinator for the East-West Center Participant Association. We asked them about their game designing experiences using the Makahiki game design interface. In addition to the interview data, Makahiki system log file and email exchanges were analyzed to identify the problems encountered during the game design process.

[Table 6.23](#) and [Table 6.24](#) lists the game design experience with Makahiki for the 2012 HPU and 2012 EWC Kukui Cup instances respectively.

One EWC designer responded that “It is easy to create the smartgrid game, putting (up) the video etc. The interface was easy to use”. She also mentioned that it is “Just a little bit time consuming”.

In summary, the general responses from the game designers of the real world Makahiki in-

Problem encountered	Cause
Error displaying the prize page after adding a prize	Makahiki software did not validate the prize parameter inputted
The introduction video referenced to previous year	The introduction video should be made generic so it could be reused over years
Error when users saved their profile in the profile page	Python Image Library was not installed correctly
Confusion of the event code generation	The admin interface to generate the event code was not intuitive
Don't know how to creating external link in Kukui Cup Activities description	The markup language for activity description is not WYSIWYG
Smartgrid layout did not change immediately after the settings changed	The cache was not clear automatically when setting changed

Table 6.23: Makahiki Game Design Experiences in 2012 HPU Kukui Cup

Problem encountered	Cause
Confused when generating the confirmation code	the admin interface was not intuitive
Don't know how to use video other than youtube	Makahiki software only supported youtube video id
Forgot to change the default commitment period	the default was meant for testing and not typical
Not able to delete the FAQ widget from help page	the admin interface to find the name of the widget on a page is not intuitive

Table 6.24: Makahiki Game Design Experiences in 2012 EWC Kukui Cup

stances are positive. There was some confusion in the usability of the game design admin interface, especially in the area of generating the event confirmation code and the user interface is not intuitive enough for easy game content creation. Both the in-lab study and post-hoc interviews revealed the confusion in the event confirmation code generation interface.

#### 6.2.4 Makahiki Game Manager Assessment

In order to gain insight into the experience of real world game managers who use the Makahiki software to manage a serious game, I performed interviews with the game managers of the two Kukui Cup challenges at Hawaii Pacific University (HPU) and East West Center (EWC) in the year 2012. The game manager for HPU is the sustainability coordinator for the Hawaii Pacific University; while the game managers for EWC are two sustainability coordinators for the East-West Center Participant Association. They are also the game designers for their Kukui Cup instances. We asked

them about their game management experiences using the Makahiki admin interface. The Makahiki instance log files and email exchanges between the game managers and us were also analyzed in order to identify the problems encountered during game management process.

**Table 6.25** shows the reported problems encountered during the game management in the 2012 HPU and EWC Kukui Cup instances.

Problem encountered	Cause	Reporting Instance
Not easy to find the event confirmation code	the admin interface to view the event confirmation code was not intuitive	HPU and EWC
Error when clicking on the "save and add another" button in the approval admin interface	the approval admin interface should remove the "save and add another" button	HPU and EWC
Some status data disappeared after the competition ended	Makahiki software bug	HPU
Missing two days worth of energy and water data	did not enter the manual energy and water data in time	EWC
Intermittently unable to access the website	there was a DNS problem with the domain name registrar	EWC
Did not support sending out email daily about the current status of the game	software not support yet	EWC
Need to make the game site available to player after the competition over	software not support yet	EWC

Table 6.25: Makahiki Game Managing Experiences in 2012 HPU and EWC Kukui Cup

When asked if it was easy to approve the player's submissions, the HPU game manager responded that it was "very easy". EWC game managers responded that "the interface was easy".

Regarding the approval process, HPU game manager responded that he did not encounter any problems and there was no need to spend a lot of time. He made sure that player submissions were either approved or rejected within 12 hours. He also discovered a useful "approve in batch" feature in the approval interface without help from the Makahiki support team. Both HPU and EWC game managers found that the "Status" game analytics page was very helpful. It was their main tool to monitor and get a sense of the status of the game.

We discussed some special issues of the game managing process in the EWC instance. The EWC instance did not use smart meters so the resource usage data had to be inputted manually by the game manager. They reported that it was a hassle to manually enter the data. The access

to the water meter device was especially not pleasant. A second issue with the EWC instance was that, because the number of players in the two competing teams were significantly different, a simple aggregation of one team’s player total points did not make sense for the team points. The energy and water consumption were also largely affected by the number of residents in the building represented by the team. A few days after the game started, the game manager realized the issues need to be addressed otherwise the competition would not be fair. We modified the Makahiki software to support the per-capita in team point calculation and update the EWC instance before the first round of the game ended. A third issue with the EWC instance was the process of player registration. Typically, player accounts in Makahiki are created by game managers before the game and do not normally change during the game. In the case of EWC instance, the team rosters were not available to the game manager so they had to use a google form to sign up players. In many cases, new players were signed up during several events during the game. The game managers had to created accounts for the new players manually in the middle of the game.

In summary, SGSEAM uncovered a few problems with Makahiki game management using the interview approach. We realized that the confidence level of this assessment approach is low because of the availability of only a few data points. An experimental study approach or performing interviews with more game managers would increase the confidence level of the assessment. Nevertheless, the interview approach provided useful insights into the real-world game management experience in this kind of serious game context.

### **6.2.5 Makahiki Developer Assessment**

We assessed the developer experience using an in-lab game development study. The UHM serious game development course, ICS691 in Spring 2013, was used for the experiment. The participating students were asked to complete the assignment to develop an enhancement to Makahiki. This involved setting up a development environment, following a tutorial to create a “Hello World” widget using Makahiki, and finally, developing an enhancement to extend the functionality of Makahiki which required five development tasks. The students were asked to submit their development source code to the public source code repository (GitHub) and write a blog post to discuss their efforts and

experiences in completing development activities. The estimated completion time was 10 to 30 hours. The students were told to complete as much as possible and if they could not complete all the activities, they should report in their blog posts what they could not complete. The blog posting should also discuss the parts of the enhancement exercise that they found easy to complete, the parts they found hard to complete, and their recommendations for how the Makahiki framework could be improved to support such enhancement activities more easily in the future.

The [Table 6.26](#) shows the results of the in-lab game development study after analyzing the blog posts and the source codes submitted by the participants.

Development Task	Number of completion	Reported Difficulty
Setup development env	8	easy
Create the “Hello World” widget	8	easy
Update score_mgr class to support groups.	7	easy
Create a group scoreboard widget.	7	hard
Create a group resource widget.	4	hard
Create a group prize widget.	2	hard
Create two group statistics widgets for the status page	1	hard

Table 6.26: Makahiki Game Development Experience, (n=8)

All 8 students reported that the first three tasks of setting up the environment, creating the “Hello world” widget and extending the score\_mgr class was easy, while the rest of the enhancements were hard. Only one student was able to successfully complete all the required enhancement developments, while most of them were able to successfully complete the reported easy tasks.

[Table 6.27](#) shows the reported problems encountered during game development. The main problem students reported was the lack of documentation for the development libraries. All students reported that they spent at least 10 hours working on the assignment. 5 students stated that they had spent more than the assignment estimated hours and had to move on without completing the assignment.

The following are the list of comments and suggestions for ways to enhance the Makahiki support to developing enhancements to the framework:

One student stated in his blog that he decided to choose the Makahiki framework to develop his own serious game because of Makahiki’s features and the possibility of reducing development

<b>Problem encountered</b>	<b>Number of participants</b>
Lacking and confusing documentation, not consistent, the screenshot does not match	8
Spent more than the assignment estimated hours	5
Don't know how to add the widgets to the Heroku instance	2
Help topic is hard to change	1
Confusion between INSTALLED_APPS and INSTALLED_WIDGET_APPS	1
Lots of files in the original Prizes folder, not sure what was needed and what was not	1
Had trouble getting the templates to redirect to the right place	1
Have to find out how to "subclassing" a widget by looking at the source code	1
Had to use stackoverflow to get answer for env and Django issues	1
Learning the code base, a lot to digest, e.g. Django, template, aggregation	1

Table 6.27: Makahiki Game Development Problems, (n=8)

1. “better documentation, add type hierarchy diagram”
2. “wished for a bit more of a high-level overview of how widgets should work”
3. “how to develop with the heroku instance is not document; this guide could use more specifics, and updating to adhere to changes in the program.”
4. “Sometimes a method returned a dictionary, but the documentation did not give the dictionary keys”
5. “have a database model diagram”

Figure 6.25: Suggestion for Makahiki Game Development support

effort by using the framework.

In summary, the SGSEAM in-lab game development study revealed significant problems with developer efficiency in Makahiki. The problems reported in this assessment are very helpful to address the areas of improvement regarding Makahiki’s support for developing game enhancements.

### 6.2.6 Assessment Report

After the above assessments were performed for each SGSEAM stakeholder in Makahiki, an assessment report was created as the final deliverable of the SGSEAM assessment of Makahiki. This report is included in [Appendix D](#). It describes the strengths of the Makahiki framework that need to be maintained, the weakness areas that can be improved, and the recommended actionable steps on how to improve from each stakeholder’s perspective.

### **6.2.7 Summary**

This section described the application of SGSEAM to assess the strengths and weaknesses of the Makahiki framework. The *in-vivo* approaches were applied to the real-world Kukui Cup challenges created by the Makahiki framework, while the *in-vitro* approaches were applied to the UHM ICS691 serious game course to assess Makahiki from the perspectives of the five serious game stakeholders identified by SGSEAM: players, system admins, game designers, game managers and game developers.

The SGSEAM assessment results were described in the format of an assessment report ([Appendix D](#)) which lists the strengths and weaknesses of Makahiki, and the recommended improvement actions. The report itemizes the areas of the Makahiki framework that need improvement from the five stakeholders' perspectives. SGSEAM revealed that the major strengths of Makahiki are the ability to create an engaging and effective serious game for an organization, and the ability to provide an easy-to-use interface for game designers and managers. SGSEAM revealed the major weaknesses of Makahiki are difficulty in developing new enhancements to extend the framework from a game developer's perspective, difficulty to integrate external services such as LDAP and email server from a system admin's perspective, and the lack of WYSIWYG content authoring tool from a game designer's perspective.

## **6.3 Discussion**

This chapter described the results of Makahiki being used to create and manage 7 real-world sustainability games by 4 organizations. These games were created in various configurations and hosting environments. The successful creation of these serious game instances provides evidence that Makahiki can be successfully tailored to the needs of these organizations. The Makahiki framework performed well in the context of university settings, approximately 1000 player population and support a certain type of smart meters, and authentication. It is unknown whether the framework is able to scale to a vastly different population such as residential, or more than 1000 players. We are actively looking for opportunities to expand Makahiki to different population and settings, to

understand the design capability of Makahiki.

Most of the Makahiki experiences were in the context of university settings with university residence hall and college students. Although we had a small pilot instance targeted at an elementary school, the experience was limited, partly due to the piloting and the effort to carry out the assessment with human subjects involving minors. The expanding of player population beyond college students may have impacts on the effectiveness and usability of the framework besides the education content. For example, elementary students may find the game interface confusing with the complicated unlock mechanism in the SmartGrid game. Expanding Makahiki into other geographical locations may require software modifications to support internationalization both in the interface and content support.

This chapter also described the application of SGSEAM to assess the strengths and weaknesses of the Makahiki framework. The *in-vivo* approaches were applied to the real-world Kukui Cup challenges created by the Makahiki framework, while the *in-vitro* approaches were applied to the UHM ICS691 serious game course to assess Makahiki from the perspectives of the five serious game stakeholders identified by SGSEAM: players, system admins, game designers, game managers and game developers. The assessment identified the strengths and weaknesses of Makahiki as a serious game framework for sustainability, and produced an improvement action report with recommendations on how to improve the framework with respects to the different serious game stakeholder's perspectives.

The application of SGSEAM to Makahiki effectively identified the strengths and weaknesses of Makahiki and provided improvement suggestions to Makahiki developers. It also revealed the strengths and weaknesses of SGSEAM as an assessment method.

One weakness of SGSEAM is that the process can be expensive and time consuming. Although the approach does not mandate you to perform all the assessment approaches proposed in SGSEAM, it is desirable to perform as many approaches as possible, depending on the resources available. In the case of applying SGSEAM to Makahiki, it required a large amount of effort to carry out different assessment approaches for each stakeholder. The data collection and data analysis were also time consuming. The fact that we were not able to convince Lucid design group to do the SGSEAM

assessment could be an indication that the cost of SGSEAM process is also high from Lucid's view.

While the comprehensive SGSEAM assessment are expensive, the design of SGSEAM does allow the flexibility of picking any one of the approaches or any one of the stakeholders, in the case of only some of the stakeholders' experiences are the main concerns from the framework developers. Because of the SGSEAM approaches are additive, the more approaches are assessed, the more confidence the assessment could be. This flexibility of choosing from a set of additive approach is a strength of the SGSEAM.

The categorization of the five serious game stakeholders is a strength of SGSEAM. It forces the assessors or the developers of a serious game framework to think in term of these categories when looking for improvements for their framework. The 3-step process of SGSEAM should also help the assessors to carry out the assessment. The three corresponding deliverables are defined in the SGSEAM process. A well designed assessment plan and an organized data repository enable a successful assessment. The improvement action report provides valuable insights to the framework developers.

When creating the assessment plan and figuring out what assessment data to collect and how to collect it, assessors or developers may discover the shortcomings of the framework in advance so that they may think of way to provide these assessment data in the framework. For example, in order to carry out the engagement metrics, a variety of player interaction data will be needed. The more player metrics can be determined, the better assessment will be, and the more insights could result.

SGSEAM player assessment approaches assess the effectiveness and engagement of the output of the serious game framework, which is the combination of the framework and the game content. A badly designed game content using a good framework may result in a suboptimal SGSEAM assessment. A good result in the SGSAEM assessment indicates the framework is correctly designed to enable a good game content design. A better effectiveness assessment approach should be investigated to identify the effect of the framework by itself.

One weakness of the SGSEAM post-hoc interview approach is the small sample size. Due to the nature of the this kind of serious game, it is common to have just one or two system admin(s),

game designer(s), and game managers(s). It is often the case that the game designers are the same as the game managers. Depending on the skill set and previous experiences of the stakeholders, their experiences with the assessed framework might have different results. For example, a sysadmin with good DB admin skill will not encounter any issue with database related installation of a framework, while a sysadmin without this skill may report difficulty. To over come this weakness, a broader interview of subjects with various background may be required, or an in-lab experiment assessment approach can be employed. On the other hand, it would be difficult or expensive to choose a group of participants with typical system admin skills. For example, in the case of the SGSEAM system admin in-lab assessment of Makahiki, a group of computer science students were chosen to participate in the experiment. The results are different from real world system admin interviews. The students had more issues in installing and configuring the database than the real world system admin, and the documentation needed was different from participants with different technical backgrounds.

Despite these limitations, the application of SGSEAM can still result in useful insights into the strengths and weaknesses of a serious game framework.

# **CHAPTER 7**

## **CONCLUSION**

This dissertation investigated the design, implementation, and evaluation of a serious game framework for sustainability called Makahiki, and a stakeholder experience based assessment method (SGSEAM) for assessing serious game frameworks. This chapter summarizes the results of the research, the contributions of the research, and possible future directions.

### **7.1 Research Summary**

This research investigates the information technology infrastructure that can support effective and efficient development of serious games for sustainability. The research includes the development of an innovative serious game framework for sustainability called Makahiki, and an assessment method called SGSEAM to access the strengths and weaknesses of IT infrastructure for serious games for sustainability from the stakeholder's perspective.

The motivation for Makahiki is to provide a low cost and better alternative to current IT infrastructure that enables the creation and managing of sustainability competitions across a variety of organizations. The design of Makahiki is inspired by current research in game design in the context of both serious games and gamification.

I led an effort over four years to design and implement the Makahiki framework. Makahiki is an open source serious game framework for sustainability, which includes a variety of common services for developing sustainability games including authentication; game mechanics such as leaderboards, points, and badges; a variety of built-in games and content focused on sustainability, a responsive user interface, cloud-based deployment, and the ability to customize to meet the needs of individual organizations.

Makahiki had been used to create and manage seven real-world sustainability games by four organizations, in different environments and with different requirements. The instances are:

1. the Kukui Cup energy challenges at the University of Hawaii at Manoa in 2011, 2012 and 2014 (UHM).

2. the HPU Kukui Cup energy challenges at the Hawaii Pacific University in 2012 and 2013 (HPU).
3. the East-West Center energy and water challenge in 2012 (EWC).
4. the Kukui Cup educational challenge at the Holy Nativity School in 2013 (HNS).

The major differences between these instances are:

- Resource collection: UHM and HPU used different metering infrastructure for automatically energy data collection. EWC collected their resource data manually, and HNS did not have energy data at all.
- UHM and HPU challenges involved only energy consumption data, the EWC challenge involved both energy and water consumption data.
- UHM and HPU provided authentication services using CAS and LDAP, while EWC used both CAS and the built-in Django internal authentication, and HNS used only the internal authentication.
- User interfaces were customized to “brand” each challenge with the logo, thematic elements, and the education contents of the sponsoring organizations.
- HPU instances were hosted in its local infrastructure, while UHM, EWC and HNS were hosted in the Heroku PaaS cloud.

I assisted the HPU system admin to install their instances in HPU’s infrastructure, and integrate with HPU’s LDAP and email server, which were found to be the most difficult tasks in HPU’s installation. I also deployed the UHM, EWC and HNS instances to the Heroku cloud and actively managed these cloud instances.

While these real world Makahiki instances provide initial evidence regarding the ability of the Makahiki framework to support sustainability games in different environments, a more rigorous evaluation of Makahiki would yield better quality insight. During my literature search, I did not find any research in the area of evaluation for serious game **frameworks**, although there exists

several assessment methods for serious games and for general purpose evaluation such as system usability. I have not found any research in the area of assessment method for assessing a serious game framework from the aspect of a full lifecycle of a serious game. I designed an assessment method called Serious Game Stakeholder Experience Assessment Method (SGSEAM), to assess a serious game framework from all the stakeholders in a serious game lifecycle.

SGSEAM describes a stakeholder perspective-based method to provide detailed insight into the strengths and weaknesses of a serious game framework. It identifies the major stakeholders whose experiences affect a serious game framework: players, system admins, game designers, game managers and game developers. For each stakeholder, multiple assessment approaches can be used, depending on the resources available. The more approaches applied, the higher one's confidence in the accuracy of the assessment results.

In my research, I applied SGSEAM to Makahiki in order to gain better insight into its strengths and weaknesses as a serious game framework. The *in-vivo* assessment approaches of pre-post effectiveness study, in-game surveys and post-hoc interviews were applied to the real-world Kukui Cup challenges created by the Makahiki framework, while the *in-vitro* assessment approaches of in-lab study used the UHM ICS691 serious game course to assess the experiences in game installation, game design and game development.

The application of SGSEAM to Makahiki has effectively identified several strengths and weaknesses of Makahiki as a serious game framework for sustainability, and produced the improvement action report to the Makahiki developers to improve the framework. SGSEAM revealed that Makahiki is capable in enabling organizations to create effective and engaging serious games for sustainability. It also revealed that the major weaknesses of Makahiki are difficulty in developing new enhancements, difficulty in integrating external LDAP and email server, and the lack of WYSIWYG content authoring tool.

## 7.2 Contributions

My research has generated several contributions, including: 1) Makahiki: A serious game framework for sustainability; 2) SGSEAM: a serious game framework assessment method; 3) Insights

into the strengths and weaknesses of Makahiki; 4) Insights into creating and running a variety of real-world serious games with Makahiki; 5) Insights into managing cloud based serious games; 6) Insights into the strengths and weaknesses of SGSEAM.

### **7.2.1 Makahiki: A serious game framework for sustainability**

The Makahiki framework described in [Chapter 3](#) contributes to the field of serious games and sustainability. Makahiki creates a unique type of serious game that combines both data on resource consumption and game content related to sustainability, represents an innovation in the areas of serious games and sustainability education.

Makahiki enables different organizations to easily create customized serious games in the context of sustainability education and behavioral change. As a framework, it includes the following features: 1) the ability to tailor system functionality to support the requirements of different organizations. 2) the ability to support sustainable resource challenges such as water, food, and waste in addition to energy. 3) the ability to extend the framework with new games and modules to support different requirements. 4) the use of HTML5/CSS3 “responsive” design techniques for support of laptop, tablet, and smart phone interfaces. 5) real-time game analytics to help assess the impact of game mechanics during challenges. 6) support for PaaS (Platform as a Service) facilities such as Heroku. This enables organizations to create and deploy challenges without obtaining physical hardware and its requisite IT support.

Makahiki is an open source system hosted on GitHub [?] with less restricted MIT open source license. It consists of over 37,000 lines of Python source code, 14,000 lines of HTML code, 17,000 lines of Javascript code, and 42,000 lines of documentation.

### **7.2.2 SGSEAM: A serious game framework assessment method**

The SGSEAM described in [Chapter 4](#) contributes to the emerging field of serious game framework evaluation research. SGSEAM describes a method for assessing serious game frameworks from the stakeholder experience perspective. The goal of SGSEAM is to identify (a) major strengths of a serious game framework, which aids the community by indicating features of the framework to

emulate, and (b) major shortcomings of the framework, which aids the community by indicating features to avoid. The benefits of SGSEAM assessment are that the developers of serious game frameworks can learn and improve from the findings of the assessment.

SGSEAM identifies the major stakeholders of a serious game that are affected by the software framework to be players, system admins, game designers, game managers and game developers. For each serious game stakeholder, SGSEAM proposes several assessment approaches such as pre-post test, in-game surveys, post-hoc interviews, and in-lab experiments. Multiple assessment approaches can be used depending on the resources available. The more approaches applied, the higher one's confidence in the accuracy of the assessment results.

A “SGSEAM User Guide” [?] was created for serious game framework developers who will be interested in using SGSEAM to assess their frameworks. I also created the “SGSEAM Assessment Guide for Lucid BuildingOS” [?] which is tailored to assessing Lucid’s serious game framework. For the purpose of applying SGSEAM to Makahiki, I created the SGSEAM plan ([Appendix B](#)), data repository ([Appendix C](#)), and improvement action report ([Appendix D](#)) for Makahiki.

### **7.2.3 Insights into the strengths and weaknesses of Makahiki**

The application of SGSEAM to Makahiki identified several strengths and weaknesses of Makahiki as a serious game framework for sustainability. The strength and weakness report and improvement action report deliverable of this research provides valuable inputs to Makahiki developers for improving the framework, thus benefiting the community and organizations that use Makahiki as a tool for sustainability education and behavior change. The strengths and weaknesses also help the serious game framework design community to learn from the strengths in Makahiki, and to avoid the shortcomings of Makahiki discovered in the SGSEAM assessment.

A “SGSEAM Improvement Action Report for Makahiki” [?] was created to documented the strengths and weaknesses of the Makahiki framework based on the finding of the SGSEAM assessment to Makahiki.

#### **7.2.4 Insights into creating and running a variety of real-world serious games**

Makahiki contributed to the creation and running of seven real-world sustainability games by four organizations. These game instances engaged over 3,770 students from 4 educational institutes (University of Hawaii at Manoa, Hawaii Pacific University, East-West Center and Holy Nativity School). Through the game interface, students learned about sustainability both online and in reality to conserve energy and water consumption and other sustainability related behaviors. The in-game survey results from the multiple UHM instances indicate that the games had positive impacts on their energy literacy, self-reported sustainability awareness and self-reported behavior changes. Both the UHM and HPU instances achieved about 30% player participation rate, which may lead to impacts to the community around the participants. The engagement metrics based on the game data from 2011 and 2012 UHM instances also indicates high levels of player engagement during the 4 week period.

These games were created in various configurations and hosting environments. The major differences involve the duration of the challenge, the population that could participate in the challenge, the type of resource(s) such as energy or water, whether they have smart meters installed, and the type of server hosting. Additional differences in requirements include the type of the authentication for participation and differences in game mechanics. The successful creation of these serious game instances provides evidence that Makahiki can be successfully tailored to the needs of different organizations.

#### **7.2.5 Insights into managing cloud based serious games**

I deployed and managed the EWC, HNS and the 2012, 2014 UHM instances in the Heroku PaaS cloud infrastructure. Although there are some hosting associated costs, it was considered cost effective in these real world instances by eliminating the need to find and maintain a local server infrastructure. It also provided the capability of infrastructure scaling down and up when needed. In the case of EWC and HNS, because these two organizations did not have enough capacity and resources to use the local hosting option, they were sponsored by Kukui Cup research grants to host their instances in the Heroku cloud infrastructure with us providing technical support.

As discussed in [Section 6.1.3](#), the cost for EWC and HNS instances are relatively inexpensive. Makahiki's support for hosting the game in the cloud lowered the barriers for these organizations to run this kind of serious game to achieve their educational and sustainability goals.

Hosting and managing the two UHM instances in the cloud provided interesting experiences with cloud-based serious games. A performance test was done to estimate the appropriate initial cloud-based resources (CPU, memory, Database) to dedicate to the game instance. During the game, a cloud monitoring service was used to monitor the performance and the utilization of resources. Depending on usage demand during the game, cloud infrastructure resources were increased or decreased without the need to stop the game.

The successful use of Makahiki in the Heroku cloud platform provided a good alternative for organizations such as EWC and HNS who have less technology capacity. It also provides greater flexibility in infrastructure scaling up and down depending on the usage demand. In the context of serious games, which are often running for a period of time and have various usage demand throughout the game period, using the cloud infrastructure is a good alternative.

#### **7.2.6 Insights into the strengths and weaknesses of SGSEAM**

The application of SGSEAM to Makahiki effectively identified the strengths and weaknesses of Makahiki and provided improvement suggestions to Makahiki developers. It also revealed the strengths and weaknesses of SGSEAM as an assessment method.

The strengths of SGSEAM includes: 1) the identification of the five categories of serious game stakeholders that are affected by a serious game framework; 2) complete coverage of serious game lifecycle than other methods; 3) a three-step process and the clearly defined corresponding deliverables helping the assessors to carry out the assessment; 4) the flexibility of choosing from a set of additive assessment approaches; 5) the step of creating the assessment plan helps to discover shortcomings of the framework from a data collection's perspective.

The weaknesses of SGSEAM include: 1) the process is expensive and time consuming; 2) the player effectiveness approaches does not assess the framework by itself, instead, it assess the combined result of the framework and game content; 3) the small sample size of the post-hoc interview

approaches; 4) difficult or expensive to carry out in-lab experiments for system admin for the real system admin skill set.

Given its limitation, the application of SGSEAM can still result in useful insights into the strengths and weaknesses of a serious game framework.

## 7.3 Future Directions

There are a variety of directions that can be pursued once this research is complete. This section discusses a variety of areas for future research.

### 7.3.1 Applying SGSEAM to other serious game frameworks

The design of SGSEAM creates a research question of what are the strengths and weaknesses of this assessment method. To better answer this question, SGSEAM could be applied to another serious game development environment. BuildingOS[?] by Lucid Design Group is such a serious game framework that is suitable for SGSEAM evaluation. Our research lab made an effort to contact Lucid Design group for the collaboration. I created the assessment plan ([Appendix E](#)) for them which we hoped would lead to the application of SGSEAM to the BuildingOS system. But due to the workload of the Lucid design group, the collaboration did not continue. It is understandable that Lucid design group, as a newly found startup company, had other urgent endeavors to pursue. Looking for another suitable serious game framework to apply SGSEAM assessment is still an ongoing research direction.

### 7.3.2 Community or Consortium to expand the content and game library

In order to provide a more engaging experience to players, it would be useful to expand the game content and add new types of games. It is not easy to maintain an active existing game content while adding new content to reflect contemporary issues. New types of games such as a 3D simulation game would provide more fun and possible better engagement, but would also require a large development effort. Building a community or consortium to expand the content and game library for Makahiki could be a more sustainable approach.

This requires a redesign of parts of Makahiki to support a library for content and games, and thus further de-couple the content from the framework.

### **7.3.3 Scale and expand Makahiki**

Most of the Makahiki instances held to date have been held in institutions of higher education, except a small pilot instance targeted at an elementary school. Expanding the use of Makahiki and scaling it to a larger population will greatly extend the impact of the games created by Makahiki.

To expand the use of Makahiki in the K-12 school environment requires suitable content tailored to the K-12 curriculum, and to different learning styles and interests. The smart grid game action content would have to be redeveloped to be appropriate for the grade level being targeted. The continuing partnership with the GreenSchools organization in the HNS pilot instance might provide more insight into this area.

Expanding Makahiki into other geographical locations, such as outside of Hawaii, will mainly involve content changing and creation. In the case of locations with different languages, the Makahiki software will need to be internationalized.

### **7.3.4 Adoption of Makahiki for collegiate sustainability competitions**

Makahiki can be easily adopted by universities to create feature-rich sustainability competitions. The universities which currently use the BuildingDashboard system from Lucid design group for their sustainability competitions can use Makahiki for free, thus gaining substantial dollar savings from not buying the commercial framework. The competitions created by Makahiki also have richer features than the BuildingDashboard.

On the other hand, creating an engaging game with Makahiki requires additional time investments to configure the game and add new educational features that is tailored to their geographically locations.

### 7.3.5 Use in other PaaS Cloud platforms

Makahiki currently only supports deployment on the Heroku cloud platform. With the recent increasing usage of different cloud infrastructure in organizations, there will be a need for organizations to run Makahiki in the same cloud platforms as their other current applications or their preferred cloud platforms other than Heroku.

Makahiki is based on Django, a python web application framework. Besides Heroku, there are other popular PaaS that support Django application, such as Amazon Elastic Beanstalk, Google App Engine, dotCloud and RedHat OpenSHIFT. Makahiki should be able to be deployed to these PaaS with some small changes in parameter configurations.

Docker [?] is an interesting open source technology that provides containers for applications to run in (almost) any infrastructure. A Docker container differs from a virtual machine in the way that it comprises only the application and its dependencies without the guest OS in the case of a virtual machine. This light weight approach provides much more portability and efficiency. [Figure 7.1](#) illustrates the difference between a Docker container and a virtual machine.

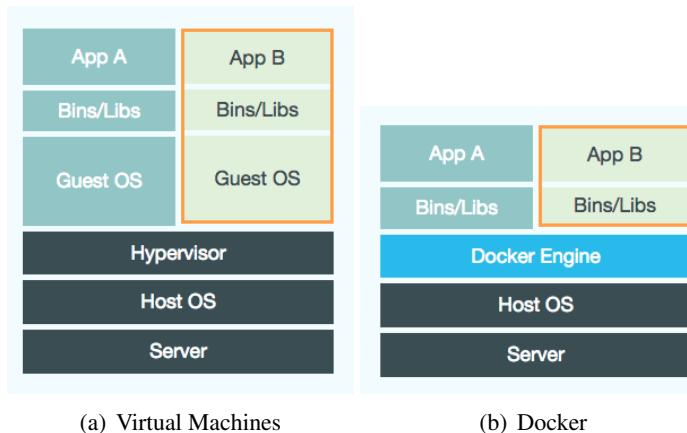


Figure 7.1: Virtual Machines V.s. Docker [?]

Makahiki could utilize Docker technology to provide a pre-configured Docker container. An organization will just download this container and deploy to any cloud infrastructure they prefer with minimum installation or configuration effort. Once deployed, the organization can take advantage of the cloud infrastructure to dynamically manage the computing resources for the Makahiki game

instance.

### **7.3.6 Enhancement to SGSEAM**

Makahiki game interface is comprised several mini-games. The current SGSEAM approach assesses the player experience of the game instance as a whole. More detailed assessment approaches targeting individual mini-games in a Makahiki instance may reveal insights on the effectiveness of the individual mini-games, thus could provide more insights for a game designer to choose or customize a mini-game.

### **7.3.7 Enhancement to Makahiki framework**

Lastly, Makahiki framework would benefit from further enhancement. Some of the enhancement projects for Makahiki that we believe would be interesting and useful for the framework include:

#### **7.3.7.1 Real-time player awareness**

Currently it is not possible in Makahiki to know who is currently online and playing the game. Creating this awareness opens up new social gaming opportunities (performing tasks together), new opportunities for communication (chat windows), and potentially entirely new games (play against another online player).

The goal of this enhancement is to extend the framework with a general purpose API that provides the identities of those who are online, and then the development of one or more user interface enhancements to exploit this capability.

#### **7.3.7.2 Deep Facebook integration**

Makahiki currently supports a “shallow” form of Facebook integration: you can request that your Facebook photo be used as your Makahiki profile picture, and you are given an opportunity to post to Facebook when the system notifies you of an accomplishment.

One way to enhance the use of Facebook is to deepen the connection between user Facebook pages and their game play. This might involve more automated forms of notification (i.e. the same

way “Spotify” playlists are posted to your Facebook wall), or ways in which your activities on Facebook could impact your Makahiki challenge status. For example, posting a sustainability video to Facebook, or liking a Sustainability organization could earn you points.

A different type of enhancement is to allow challenge designers to specify a Facebook page as the official Challenge Facebook information portal, and have the system automatically post information to that Facebook page as the challenge progresses.

#### **7.3.7.3 Customization of game analytics page**

Makahiki provides a one-page game analytics information for game designers and managers to monitor the game and get insights into the current status of a running game. The page currently crams 19 analytics widgets into one page, demanding a very large screen real estate to view the information in whole or the need of frequent scrolling.

One enhancement is to provide admin users an additional interface to customize the appearance of the analytic widgets where they can choose which widgets appear. The individual widget could also support two types of appearance, one has a larger display that provides full information and one has a smaller display that provides the minimum essential information.

#### **7.3.7.4 Action Library Management System**

Makahiki currently ships with over 100 possible “actions” already developed for the Smart Grid Game. However, the current implementation suffers from a number of problems:

There is no convenient way to display and peruse the current set of actions. This has led to a duplicate representation of the smart grid game, implemented using a Google Docs spreadsheet linked to Google Sites pages. This approach has a lot of problems: it duplicates content, and it does not provide a way to edit or manage content, it is already out of date.

The content is intimately tied to the Smart Grid Game implementation. The SGG is just one of many ways that the sustainability content could be presented to players. By separating “content” from the “presentation”, more games can be developed using this content. We could enhance Makahiki to provide a “content management system” for “actions”, which could involve the follow-

ing changes to the current Makahiki system:

- Library actions are created, viewed, and edited via a separate admin interface which is defined for the content authoring role
- An editor is provided to create action content and preview it in a formatted manner.
- Library actions can be “instantiated” into the Smart Grid Game.
- Library content can be exported and imported into systems in order to support sharing.
- A public repository can be provided on GitHub. The format could be JSON.

## APPENDIX A

### PUBLICATION LIST

These are the publications that have come out of this research that I have authored or co-authored:

#### A.1 Journal Paper

- Robert S. Brewer, **Yongwen Xu**, George E. Lee, Michelle Katchuck, Carleton A. Moore, Philip M. Johnson. Three Principles for the Design of Energy Feedback Visualizations. In *International Journal On Advances in Intelligent Systems*, Vol. 3 & 4, No. 6. (2013), pp. 188-198

#### A.2 Conference Papers

- **Yongwen Xu**, Philip M. Johnson, George E. Lee, Carleton A. Moore, Robert S. Brewer. Makahiki: An open source serious game framework for sustainability education and conservation. In *Proceedings of the 2014 International Conference on Sustainability, Technology, and Education (STE 2014)*, Taiwan, December 2014
- **Yongwen Xu**, Philip M. Johnson, Carleton A. Moore, Robert S. Brewer, Jordan Takayama. SGSEAM: Assessing serious game frameworks from a stakeholder experience perspective. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications (Gamification 2013)*, Ontario, Canada, October 2013
- Robert S. Brewer, **Yongwen Xu**, George E. Lee, Michelle Katchuck, Carleton A. Moore, and Philip M. Johnson. Energy feedback for smart grid consumers: Lessons learned from the Kukui Cup. In *Proceedings of the Third International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies (ENERGY 2013)*, Lisbon, Portugal, March 2013.

- Philip M. Johnson, **Yongwen Xu**, Robert S. Brewer, Carleton A. Moore, George E. Lee, and Andrea Connell. Makahiki+WattDepot: An open source software stack for next generation energy research and education. In *Proceedings of the 2013 Conference on Information and Communication Technologies for Sustainability (ICT4S)*, Zurich, Switzerland, February 2013.
- Philip M. Johnson, **Yongwen Xu**, Robert S. Brewer, George E. Lee, Michelle Katchuck, and Carleton A. Moore. Beyond kWh: Myths and fixes for energy competition game design. In *Proceedings of Meaningful Play 2012*, October 2012.

### A.3 Workshop

- Robert S. Brewer, George E. Lee, **Yongwen Xu**, Caterina Desiato, Michelle Katchuck, and Philip M. Johnson. Lights Off. Game On. The Kukui Cup: A dorm energy competition. In *Proceedings of the CHI 2011 Workshop on Gamification*, Vancouver, Canada, May 2011.
- **Yongwen Xu**. Designing a Serious Game Framework for Sustainability. In *PhD Student Workshop on ICT4S 2013*, Zurich, Switzerland, February 2013.

### A.4 Poster

- Robert S. Brewer, Philip M. Johnson, Michelle Katchuck, George E. Lee, **Yongwen Xu**. Lights Off. Game On. The 2011 Kukui Cup. In *The Behavior, Energy and Climate Change Conference (BECC) 2011*, Washington DC, November 2011.

## APPENDIX B

### SGSEAM PLAN FOR MAKAHIKI

This appendix includes the SGSEAM assessment plan for Makahiki. It is the deliverable for the first step of SGSEAM when applying to Makahiki framework. It first identifies the stakeholders, determines the appropriate assessment approaches according to the available resources, chooses assessment participants, and creates the assessment schedule.

#### B.1 Identify SGSEAM Stakeholders in Makahiki

The first step in SGSEAM assessment plan is to identify the stakeholders in Makahiki. [Table B.1](#) listed the identified stakeholders who use the Makahiki framework.

Stakeholder class	Tasks	Role
Player	Participate in the Makahiki games	Students living in the residential halls
System admin	Install Makahiki software, monitor and scale the system, backup, patch maintenance	IT staffs
Game designer	Design the content, configure suitable games and mechanics	Challenge organizers
Game manager	Manage the game during the period of game play.	Challenge organizers
Developer	Develop customization, extend and enhance the game and framework.	Makahiki developers

Table B.1: SGSEAM Stakeholders

#### B.2 Determine SGSEAM Approaches for Makahiki

The second step in SGSEAM plan is to determine the assessment approach. As described in SGSEAM, approaches include both *in-vivo* and *in-vitro* assessments. The *in-vivo* approaches, such as pre-post test, in-game surveys and post-hoc interviews, assess the real world instance of the game. The *in-vitro* approaches use in-lab experiments in a simulated environment. Different assessment approaches will have different levels of rigor or validity. When applying SGSEAM in Makahiki, I

used the real world Makahiki instances as the in-vivo approaches which includes pre-post effectiveness study for player assessment, post-hoc interview for game administrator and game designer.

In addition to real world instances assessment, I also implemented the in-vitro assessment approach using in-lab experiments. In Spring 2013, Professor Philip Johnson at the Information and Computer Science Department of University of Hawaii used Makahiki to teach a course in serious game development. The students were seniors or graduate students majoring in computer science related fields. During the course, the students installed Makahiki, designed a serious game instance with Makahiki, and developed an enhancement to the Makahiki system. The participation was voluntary. This is considered as an in-lab experiment since they are evaluating Makahiki in a class setting and using Makahiki in the development environments.

**Table B.2** lists the SGSEAM approaches that are used to assess the strengths and weaknesses of Makahiki from different stakeholders' view.

Stakeholder	Assessment approaches	Expected Outcomes
Player	Pre-post effectiveness study	Determine effectiveness in energy literacy and resource usage reduction
	Self-reported effectiveness survey	Determine self-reported effectiveness in behavior change and awareness
	Self-reported usability survey	Identify problem areas in game interface
	Engagement metrics	Determine the extent of engagement
System admin	Post-hoc admin interview	Determine strengths and weaknesses in system install and maintenance
	In-lab system admin study	
Game designer	Post-hoc designer interview	Determine strengths and weaknesses in facilitating the game design process
	In-lab game design study	
Game manager	Post-hoc manager interview	Determine strengths and weaknesses in managing the game
Developer	In-lab game development study	Determine strengths and weaknesses in developing system enhancement

Table B.2: SGSEAM approaches

The following sections describe the assessment approaches in details.

### B.2.1 Player Assessment Plan

I plan to use the real-world Makahiki instances (UHM KC 2011, 2012, 2014) at the University of Hawaii at Manoa to study the player's experience with the Makahiki framework. There will be over

1000 eligible players for each of these instances. The players are first year college students living in four similar structured residence halls in close vicinity. There are smart electrical meters installed in these residence halls.

To assess the effectiveness of the framework for designing games that improve player literacy in sustainability, we will conduct two energy literacy surveys, one before the challenge (pre-game) and one after the challenge (post-game). SurveyGizmo will be used to create the surveys which consists of the set of sustainability literacy and behavior questionnaires. The response from the two surveys will be analyzed to provide insight about the player's literacy and behavior change.

To assess the effectiveness of the framework for designing games that produce positive change in sustainability behaviors, we will record and analyze the energy consumption data before, during and after the challenge. Before the challenge, an energy usage baseline will be established. The energy consumption data will be examined to understand any usage pattern or reduction during and after the challenge. We will also conduct an in-game self-reported behavior change survey. The survey will ask questions about player interests in sustainability prior to and after the game, as well as any perceived behavior changes when playing the game. The in-game survey questionnaires are described in [Appendix F](#).

To assess the usability of the game produced by the Makahiki framework, we will conduct an in-game usability survey. The survey will ask questions about the players' experience with respect to the user interface of the game. The response from the survey will be analyzed to provide insight about the game usability.

In addition to the surveys and energy data measurement, the following engagement metrics will be calculated based on the game and log data to assess the engagement level of the instance:

- Participation rate
- Number of players per day
- Play time per day
- Submissions per day
- Social interactions per day

- Website errors per day

### B.2.2 System Admin Assessment Plan

I plan to use two approaches to assess the system admin's experience: One is an in-lab experiment, the other is interviews with the system admin of a real world Makahiki instance.

I will conduct an in-lab experiment with the students in the ICS691 (a serious game development class in the Department of Computer Science at UHM) in Spring 2013. The students will be tasked with installing the Makahiki system into their local computers as well as the cloud environment. In order to understand how much time it takes to install the Makahiki and what problems might be encountered, I designed a Google form which details the steps for installing Makahiki both locally and in the cloud. I will ask students to use this form to record the time they spent and the problems they encountered during each step.

[Figure B.1](#) illustrates a partial google form used for Makahiki system admin assessment. [Appendix G](#) includes the complete google form.

The students will also be asked to provide feedback about their installation experiences in the form of a blog post. In the blog post, I will ask them to discuss the following topics:

- What is the most difficult step during installation?
- What problems did you encounter during the installation?
- Have you install any database, web server or similar server products prior to this assignment?  
Are those installations for development or production purpose?
- If you have experience installing other servers before, How does your prior experience of installing other servers compare to the installation of Makahiki?
- What could be improved about the Makahiki installation process?
- Compare your experience of installing Makahiki in Heroku with installing it locally,

The data collected from the Google form responses and the blog posts from the students will be analyzed to gain insight into how easy or difficult it is to install Makahiki.

## 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

### 2.1.1.1.2. Install Python \*

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

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

### 2.1.1.1.3. Install C Compiler \*

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

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

Figure B.1: Makahiki Developer assessment Form

In order to gain insight into the experience of a real world system admin who uses Makahiki, I will perform interviews to the system admin of the 2012 Hawaii Pacific University (HPU) challenge. The interview questions will include:

- How much time did you spend to install the Makahiki system?
- How much time did you spend to maintain the Makahiki system, including backup, monitoring?
- What problems did you encounter during your installation and administering the Makahiki system?

I will also collect any email exchanges with the system admin regarding the installation process and maintenance of Makahiki system. Both interview and email exchange data will be analyzed to

understand the system admin's experiences with Makahiki.

### B.2.3 Game Designer Assessment Plan

Similar to Makahiki system admin assessment, I plan to use two approaches to assess the game designer's experience: One is an in-lab experiment, the other is interviews with the game designers of real world Makahiki instances.

I will conduct an in-lab experiment with the same students in the ICS691 class. The students will be tasked to design a Kukui Cup-like serious game using Makahiki. I designed another Google form with the detailed steps for designing a game in Makahiki. I will ask students to follow these steps and record their time and problems encountered during their designing process. [Appendix G](#) has the complete Google form for the steps the students need to follow.

In addition, the students will be asked to provide feedback about their game design experiences in the form of blog posts that discuss the following topics:

- What is the most difficult step during Challenge Design?
- What problems did you encounter while designing the challenge?
- What could be improved in the Makahiki Challenge Design process?

The data collected from the responses of the game design Google forms and the blog posts from the students will be analyzed to gain insight into the game design experiences with Makahiki.

In order to gain insight into the experience of real world game designers who use Makahiki to design serious games, I will perform interviews with the real-world game designers of the 2012 Hawaii Pacific University challenge and the 2012 East West Center challenge. I will asked them about their experiences using the Makahiki game design admin interface to design their sustainability serious games. The interview questions will include:

- How much time did you spend to configure the challenge global settings?
- How much time did you spend to design the individual games?
- What problems did you encounter?

- Did you find it difficult to design a specific game? which one, what was difficult?
- What did you like the least when using the system?

I will also collect any email exchanges with the game designers regarding the game design process. Both the interview and email exchange data will be analyzed to gain insight into the game design experiences with Makahiki.

#### **B.2.4 Game Manager Assessment Plan**

I plan to perform interviews with the real world game managers of the 2012 Hawaii Pacific University challenge and 2012 East West Center challenge to study the experience of game management using Makahiki. The interview questions will include:

- How much time did you spend to approve the action submissions?
- How much time did you spend to monitor the game status?
- What problems did you encounter?
- Did you find it difficult to manage? what was difficult?
- What did you like the least when using the system?

I will collect any email exchanges with the game managers. The data collected from the interviews and email exchanges will be analyzed to gain insight into the game managing experiences with Makahiki.

#### **B.2.5 Developer Assessment Plan**

I plan to perform an in-lab game development study experiment with the students participating in the ICS691 serious game development class in Spring 2013. The students will be tasked with developing an enhancement to the Makahiki instance. This will involve setting up the development environment, following the tutorial to create the “Hello world” widget using Makahiki, and finally, developing an enhancement which extends the functionality of the Makahiki system. The enhancement is specified in 5 development tasks.

The students will be asked to submit their development source code to the public source code repository (Github) and write a blog post to discuss their efforts in completing the development activities. The blog post should discuss the following topics:

- What part is complete?
- What part is not complete?
- Which parts you found easy or hard to complete?
- What problems did you encounter while developing this enhancement tasks?
- What is your recommendations for the framework to improve development support.

I will review their source code to compare their code to the reference implementation, analyze the blog posts, as well as any email correspondences regarding the problems encountered during the development.

### B.3 Choose Assessment Participants

After the assessment approaches are determined, the next step in SGSEAM is to identify the assessment participants for the different stakeholders. **Table B.3** lists the participants for assessing the Makahiki framework using SGSEAM.

Stakeholder class	Person(s)	Organization
Player	All eligible players in the UH KC instance	UHM
System admin	ICS691 students, system admin for HPU instance	UHM, HPU
Game designer	ICS691 students, game designers for HPU & EWC instance	UHM, HPU, EWC
Game manager	ICS691 students, game managers for HPU & EWC instance	UHM, HPU, EWC
Developer	ICS691 students	UHM

Table B.3: SGSEAM Participants for Makahiki

## B.4 Create Assessment Schedule

After determining what the assessment approaches and who the participants are, the next step is to create the assessment schedule. The schedule for applying SGSEAM to Makahiki is shown in

**Table B.4.**

Time	Task	Assessment
Fall 2011	Pre-post and in-game surveys, engagement metrics collection with UHM 2011 KC	Player experience
2012	In-game survey, engagement metrics collection with UHM 2012 KC	Player experience
Fall 2012	Interview with HPU KC sysadmin	System admin experience
Fall 2012	Interview with HPU & EWC KC game designers	Game designer experience
Fall 2012	Interview with HPU & EWC KC game managers	Game manager experience
Spring 2013	In-lab installation experiment with UHM ICS691 students	System admin experience
Spring 2013	In-lab game design experiment with UHM ICS691 students	Game designer experience
Spring 2013	In-lab game development experiment with UHM ICS691 students	Developer experience
Spring 2014	in-game survey with UHM 2014 KC	Player experience

Table B.4: SGSEAM Assessment Schedule for Makahiki

## APPENDIX C

### SGSEAM DATA REPOSITORY FOR MAKAHIKI

This appendix includes the descriptions of the data collected in the data repository from the process of applying SGSEAM assessment to Makahiki. It is the second deliverable of the SGSEAM process.

The data in the repository are divided into five categories according to the five stakeholders: player, system admin, game designer, game manager, and developer. They are organized into five corresponding folders in a portable hard drive under the main folder called “SGSEAM-Makahiki-Data-Repository”. The folders were backed up periodically. Because of the sensitive information in the data, the data repository is currently only accessible by me.

During the SGSEAM assessment process, data were collected and stored into the data repository according to their categories.

#### C.1 Player Assessment Data

According to the SGSEAM assessment plan for Makahiki, the player assessment includes 4 approaches, which applied to the UHM Kukui Cup instances from 2011, 2012 and 2014. The data collected during the assessment process were stored in the “player” folder under the main data repository folder. The player assessment data are described in [Table C.1](#).

Data	Format	Filename	Size
Pre and post surveys from UHM 2011 KC	CSV (SurveyGizmo export)	2011-preGameSurvey.csv, 2011-postGameSurvey.csv	120KB
Energy data before, during, and after UHM 2011 KC	DUMP (Wattdepot DB dump)	2011-wattdepotDB.tgz	180MB
In-game effectiveness and usability survey from UHM 2011, 2012, 2014 KC	CVS/XLS (SurveyGizmo export)	2011-inGameSurvey.xls, 2012-inGameSurvey.csv, 2014-inGameSurvey.xls	259KB
Game data from UHM 2011, 2012, 2014 KC	DUMP (Makahiki DB dump)	2011-makahikiDB.dump, 2012-makahikiDB.dump, 2014-makahikiDB.dump	205MB

Table C.1: Player Assessment Data

## C.2 System Admin Assessment Data

The system admin assessment data are described in [Table C.2](#). The data files are stored in the “sysadmin” folder.

Data	Format	Filename	Size
Interviews audio recordings from HPU sysadmin	WAV	HPU-SysadminInterview.wav	141.9MB
Email communications with HPU sysadmin	PDF (Gmail archive)	gmailArchiveSysadmin.zip	2.2MB
Google form responses from ICS691 installation experiment	XLS (Google Form export)	herokuInstallLog.xlsx localInstallLog.xlsx	41KB
Blog posts from ICS691 installation experiment	PDF (blog archive)	herokuInstallBlogArchive.zip, localInstallBlogArchive.zip	4.6MB

Table C.2: Sys admin Assessment Data

## C.3 Game Designer Assessment Data

The game designer assessment data are described in [Table C.3](#). The data files are stored in the “game-designer” folder.

Data	Format	Filename	Size
Interviews audio recordings from HPU & EWC game designers	WAV (archive)	HPU-designerInterview.zip, EWC-designerInterview.zip	534MB
Email communications with HPU & EWC game designers	PDF (Gmail archive)	HPU-gmailArchiveDesigner.zip, EWC-gmailArchiveDesigner.zip	1.8MB
Google form responses from ICS691 game design experiment	XLS (Google Form export)	designLog.xlsx	19KB
Blog posts from ICS691 game design experiment	PDF (blog archive)	designBlogArchive.zip	1.8MB

Table C.3: Game Designer Assessment Data

## C.4 Game Manager Assessment Data

The game manager assessment data are described in [Table C.4](#). The data files are stored in the “game-manager” folder.

Data	Format	Filename	Size
Interviews audio recordings from HPU & EWC game managers	WAV	HPU-managerInterview.wav, EWC-managerInterview.wav	137MB
Email communications with HPU & EWC game managers	PDF (Gmail archive)	HPU-gmailArchiveManager.zip, EWC-gmailArchiveManager.zip	109KB

Table C.4: Game Manager Assessment Data

## C.5 Developer Assessment Data

The developer assessment data are described in [Table C.5](#). The data files are stored in the “developer” folder.

Data	Format	Filename	Size
Blog posts from ICS691 development experiment	PDF (blog archive)	developerBlogArchive.zip	16.3MB

Table C.5: Developer Assessment Data

## **APPENDIX D**

### **SGSEAM IMPROVEMENT ACTION REPORT FOR MAKAHIKI**

This appendix describes the improvement action report after applying the SGSEAM assessment to Makahiki. It is the deliverable for the final step of the SGSEAM process. It first describes findings from the data analysis of the SGSEAM assessment for each stakeholder, followed by a list of the strengths and weaknesses of the Makahiki framework from the perspectives of different stakeholders, and suggestions of any improvement actions to the framework.

#### **D.1 Makahiki Player Assessment Report**

##### **D.1.1 Player Assessment Results**

Following the SGSEAM assessment plan for Makahiki ([Appendix B](#)), four assessment approaches were carried out to assess the player's experiences with the Makahiki framework. The real world Makahiki instances of the UHM Kukui Cup challenge were used for these assessments.

The Pre-post effectiveness study shows that the players of the Makahiki framework effectively achieved the energy literacy improvement. Since the average energy reduction found in the study was low, the study did not provide strong evidence that the challenge results in effective reduction of energy consumption by players.

The self-reported effectiveness survey study shows that the players perceived their experiences with Makahiki were positive and there were impacts to their sustainability awareness and behaviors because of playing the game.

The self-reported usability survey study reveals that most of the usability in Makahiki is good. It discovered a few usability issues in the 2014 KC instance such as the slow performance of certain pages and the confusion of scoreboard display. There were also issues with the SmartGrid game content which had several broken links causing players not able to complete the actions.

The engagement metrics study indicates that Makahiki is successful in achieving player engagement for a 4 week period in this kind of serious game. After this period, engagement tampered off.

### **D.1.2 Strengths and Weaknesses from Player's Perspective**

From the above player assessment results, we concluded that the strengths of the Makahiki framework from player's experience perspective as listed below:

1. The framework can be successfully used to create a game that effectively improve player's energy literacy.
2. The framework can be effective to increase player's awareness and behavior change in sustainability.
3. The framework can create a series of games with good usability.
4. The framework can have a high level of player engagement for a duration of 4 weeks.

We found the weaknesses of the Makahiki framework from the player's experience perspective are:

1. The games did not show effective reduction of energy consumption.
2. Player engagement is low for a duration that is longer than 4 weeks.
3. The rotating carousel display of the scoreboard widget is confusing to some players.
4. There seems to be a performance downgrade in the latest software release.
5. There is no tool to validate the game content such as the broken links

### **D.1.3 Improvement Actions from Player's Perspective**

Based on the findings of the weaknesses of the framework, we recommend several improvement actions to the Makahiki framework from the player stakeholder's perspective, as listed in [Table D.1](#).

#### **Improvement Actions:**

1. Investigate other methods to measure the effectiveness of resource consumption.
2. Research the benefits and approaches to maintain high level of engagements for long competition duration.
3. Investigate the performance downgrade in the latest release.
4. Investigate alternative to improve the rotating scoreboard widget display.
5. Fix the broken links to the educational video.
6. Provide a content validation tool to minimize content related issues.

Table D.1: SGSEAM Improvement Action Report from Player's Perspective

## **D.2 Makahiki System Admin Assessment Report**

### **D.2.1 System Admin Assessment Results**

Two SGSEAM assessment approaches were carried out to assess the system admin's experiences with the Makahiki framework.

The in-lab installation experiment study identified that the “Install and configure database” step of the local installation took the longest average time. It also has the most problems reported from the participants. In the case of cloud installation, the study identified that the “Setup cloud environment” step took the longest average time. The most participant-reported problems were confusions in setting up Amazon S3. It also identified a few issues with the installation documents and the diagnosability with the install script when errors occurred.

The real-world system admin interview study revealed difficulty in integrating Makahiki with an organization's LDAP and email server as well as using SSL in a local installation scenario. The real-world system admin reported no issues in maintaining the Makahiki system with backup and monitoring once the installation was completed.

The study show that the installation experiences varied greatly between the participants in the study. For the participants that may have better system admin skill, it only took 0.9 hour to install the system locally and 0.7 hour to install in the cloud. This indicated Makahiki is relatively easy to install in both scenarios for experienced system admin.

### **D.2.2 Strengths and Weaknesses from System Admin's Perspective**

From the above results of the system admin assessments, we can conclude that the strengths of the Makahiki framework from system admin's experience perspective are:

1. The framework is easy to install for experienced system admin.
2. Cloud installation is easier than local installation for experienced system admin.
3. Easy to maintain the system once the installation is completed.

We found the weaknesses of the Makahiki framework from the system admin's experience perspective are:

1. Database installation documentation is not easy to follow for inexperienced database admins.
2. It is not easy to diagnose problems from the output of the installation script when problems occur.
3. It is difficult to integrate with an organization's LDAP and email server
4. It is difficult to use SSL with the Makahiki server

### **D.2.3 Improvement Actions from System Admin's Perspective**

Based on the findings of the weaknesses of the framework, we recommend several improvement actions to the Makahiki framework from the system admin stakeholder's perspective, as listed in [Table D.2](#).

<b>Improvement Actions:</b>
1. Improve the database configuration document.
2. Improve automation of DB installation, configuration, cloud environment setup process as much as possible.
3. Improve the usability of the installation script, with clear error message, redirecting the verbose output into installation log.
4. Develop a tool to verify and diagnose the dependency and environment setup.
5. Improve the documentation on LDAP and email server integration.
6. Improve the documentation on configuring SSL.

Table D.2: SGSEAM Improvement Action Report from System Admin's Perspective

## **D.3 Makahiki Game Designer Assessment Report**

### **D.3.1 Game Designer Assessment Results**

Two SGSEAM assessment approaches were carried out to assess the game designer's experiences with the Makahiki framework.

The in-lab game design study revealed the two most time consuming tasks in Makahiki design were “Smart Grid Game Design” and “Configure Challenge Settings”. The complexity of the predicate system and lack of documentation caused difficulty in defining dependencies between game activities when designing the SmartGrid game. There was also difficulty encountered when gener-

ating attendance codes for the event activities. The study discovered a few bugs in the game design interface when configuring the challenge settings.

The real world game designer interview study also revealed the confusions in the game design interface for generating the event attendance code. It was reported that the user interface is not intuitive enough for easy game content creation, and it is time consuming to design the SmartGrid game.

### D.3.2 Strengths and Weaknesses from Game Designer's Perspective

From the above results of the game designer assessments, we can conclude that the strengths of the Makahiki framework from game designer's experience perspective are:

1. Most of the game design interface are easy to use.
2. It is easy to create a game with small modification to the default content.

We found that the weaknesses of the Makahiki framework from the game designer's experience perspective are:

1. The predicate system is difficulty and lack of documentation.
2. It is difficulty to generate event attendance code.
3. Content creation is not WYSIWYG.
4. Designing the smart grid game is time consuming.

### D.3.3 Improvement Actions from Game Designer's Perspective

Based on the above findings, we recommend several improvement actions to the Makahiki framework from the game designer stakeholder's perspective, as listed in [Table D.3](#).

#### Improvement Actions:

1. Improve usability in event confirmation code generation admin interface
2. Provide an WYSIWYG content authoring and design tool
3. Improve the documentation on how to use the predicates
4. Investigate a new design of predicate system to improve usability
5. Fix the reported bugs in the challenge setting configuration interface

Table D.3: SGSEAM Improvement Action Report from Game Designer's Perspective

## **D.4 Makahiki Game Manager Assessment Report**

### **D.4.1 Game Manager Assessment Results**

The post-hoc game manager interview approach was carried out for the game manager assessment. The real-world game manager interview study uncovered a few problems with Makahiki game management. The reported problems included the confusions in finding the event confirmation codes from the game managing interface, and not able to automatically send out game status emails. The interview study show that the submission approval interface is easy to use, and the game analytics page is very helpful when managing the game.

### **D.4.2 Strengths and Weaknesses from Game Manager's Perspective**

From the above results of the system admin assessments, we can conclude that the strengths of the Makahiki framework from game manager's experience perspective are:

1. The submission approval interface is easy to use.
2. The batch approval feature is useful.
3. The game analytics "Status" page is very useful.

The weaknesses of the Makahiki framework from the game manager's experience perspective are:

1. It is not easy to find the event confirmation code from the admin interface
2. The game site is not available after the competition is over
3. The framework do not support automatically sending out game status emails

### **D.4.3 Improvement Actions from Game Manager's Perspective**

Based on the above findings, we recommend several improvement actions to the Makahiki framework from the game manager stakeholder's perspective, as listed in [Table D.4](#).

**Improvement Actions:**

1. Improve usability of the interface for looking up event confirmation code
2. Support access to the game site after the game over
3. Support automatic emailing of game status periodically

Table D.4: SGSEAM Improvement Action Report from Game Manager's Perspective

## D.5 Makahiki Developer Assessment Report

### D.5.1 Developer Assessment Results

The in-lab game development approach was carried to assess the developer's experience with Makahiki. The study revealed many problems with the development support in Makahiki. The most reported problem was the lack of documentation regarding the development with Makahiki. The simple "HelloWorld"-type development was well documented, but beyond that, the documentation was lacking. There was a steep learning curve to develop with the framework.

### D.5.2 Strengths and Weaknesses from Developer's Perspective

The strengths of the Makahiki framework from developer's experience perspective are listed here:

1. The framework can be used to develop other serious games with less effort
2. It is easy to develop a simple enhancement.

The weaknesses of the Makahiki framework from the game designer's experience perspective are:

1. It is difficult to develop complex enhancements with the current documentation.
2. Some documents are confusing and inconsistent.
3. The learning curve for development is steep.

### D.5.3 Improvement Actions from Developer's Perspective

Based on the above findings, we recommend several improvement actions to the Makahiki framework from the developer stakeholder's perspective, as listed in [Table D.5](#).

**Improvement Actions:**

1. Create documentation about widget development overview, data models, class diagrams.
2. Improve documentation on existing structure, modules, APIs.
3. Include more examples or tutorials in developing with the framework
4. Investigate the plugin development mechanism

Table D.5: SGSEAM Improvement Action Report from Developer's Perspective

## APPENDIX E

# SGSEAM GUIDE FOR LUCID BUILDING OS AND BUILDING DASHBOARD

This appendix includes the SGSEAM assessment guide written specifically for BuildingOS and BuildingDashboard, with the intention of administering SGSEAM assessment to the Lucid Design Group's serious game framework. Although the assessment was not actually carried out, this guide illustrates an example for SGSEAM assessment to a similar serious game framework.

### E.1 Overview

This document describes how to assess the Lucid BuildingOS and BuildingDashboard using the Serious Game Stakeholder Experience Assessment Method (SGSEAM).

The goal of this assessment is to identify the major strengths and shortcomings of the software framework using the perspectives of major stakeholders.

The cost of this assessment to Lucid is the requirement for various stakeholders to be available to me for approximately one 30 minute interview.

The benefit of this assessment is the identification of actionable improvements to Lucid BuildingOS and BuildingDashboard.

The SGSEAM assessment method is being developed as part of my Ph.D. research at the University of Hawaii. The assessment of Lucid BuildingOS and BuildingDashboard will help me to identify strengths and weaknesses in SGSEAM. All data about the Lucid BuildingOS or BuildingDashboard systems revealed through this assessment will be kept confidential and will not be presented in my research findings.

Table E.1 outlines the steps of the process of applying SGSEAM to a framework.

1. Step one is to **Plan the assessment**, including identifying the stakeholders, determining assessment approaches, and creating the assessment schedule. The deliverable for this step is the **assessment plan** document.

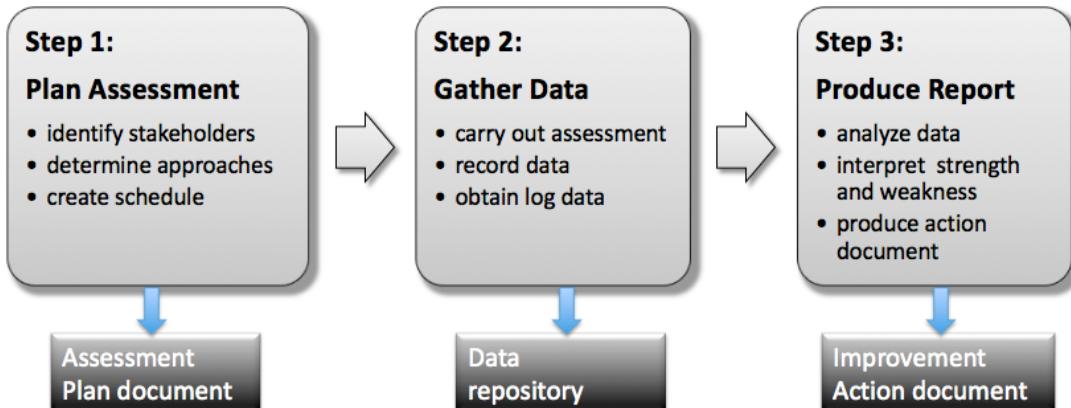


Table E.1: Applying SGSEAM to a framework

2. Step two is to **Gather data** by carrying out the assessment, recording and obtaining related data. The deliverable for this step is the assessment ***data repository***.
3. Step three is to **Produce the assessment report** by analyzing the data and interpreting strengths and weaknesses. The deliverable for this step is the ***improvement action*** document.

The following chapters describe the steps in detail. The Appendix provides additional background material. Each chapter concludes with an “Action Item” shade box, which indicates what you need to do. For example:

**Action Item:** Read the next three chapters of this document, and determine if this proposed evaluation is feasible. If you identify obstacles, please note them in the spreadsheet so that we can discuss them in an upcoming phone call.

## E.2 Step 1: Plan the Assessment

### E.2.1 Identify Stakeholders

The first step is to identify the SGSEAM stakeholders and their tasks for the Lucid BuildingOS and BuildingDashboard framework.

According to Campus Conservation Nationals (CCN) Competition Planning Guide, a Competition Organizing Team (COT) will plan and execute the competition. Besides being residents of

SGSEAM Stakeholder	BuildingOS Users	Tasks	Comments on stakeholder definitions
Player	Building resident	Use BuildingDashboard to view building data and participate in the competition.	
System admin	Internal system admin or developer	Install software, backup, patch, monitor and scale the system.	
Game designer	Behavior Change Manager, Technical Manager, Competition Director, Research Manager	<ul style="list-style-type: none"> <li>Decide on a competition format/structure</li> <li>Set up buildings, meters, and competition in BuildingOS</li> </ul>	
Game manager	Technical Manager, Marketing Manager, Building Captain Manager, Events Manager	<ul style="list-style-type: none"> <li>Collect, verify baseline and competition data, enter into BuildingOS</li> <li>Kick-off and other events</li> <li>Coordinate competition prizes</li> <li>Manage social media</li> <li>Monitor competition status</li> </ul>	
Game developer	Internal or external Developer	<ul style="list-style-type: none"> <li>Develop interface to support other meters</li> <li>Customize dashboard interface</li> </ul>	

Table E.2: BuildingOS Stakeholders

buildings participating in the competition, they are also users and stakeholders of the BuildingOS framework.

We have converted COT roles into SGSEAM stakeholders and identified their tasks related to BuildingOS and BuildingDashboard, as shown in [Table E.2](#).

**Action Item:** Review the “Stakeholders” tab in the attached spreadsheet, and provide comments if you believe the set of stakeholders or the mapping needs modification.

## E.2.2 Determine Assessment Approach

There are several possible assessment approaches for each stakeholder. Different assessment approaches have different levels of rigor which impacts upon the quality the assessment result. They also require different levels of implementation costs or efforts. [Table 4.2](#) describes the SGSEAM assessment approaches we have developed for each stakeholder category.

While an in-lab experiment has the most rigor, we believe it is too expensive for this assessment. We therefore recommend an interview approach for all stakeholders except players. [Table E.3](#) shows the approaches we recommend for each stakeholder in the case of Lucid BuildingOS and BuildingDashboard.

The following sections describe in detailed the assessment approaches for each stakeholder. description of the recommended approaches.

Stakeholder	Assessment Approaches	Expected Outcomes	Comments on approach
Player	Pre-post effectiveness study	Determine effectiveness in resource usage reduction.	
	Usability survey	Identify problem areas in game interface	
	Engagement metrics	Determine the extent of engagement	
System admin	Post-hoc admin interview	Identify strengths and weaknesses in the installation and maintenance process.	
Game designer	Post-hoc designer interview	Determine strengths and weaknesses in the game design interface.	
Game manager	Post-hoc manager interview	Determine strengths and weaknesses in the game managing interface.	
Game developer	Post-hoc developer interview	Determine strengths and weaknesses in developing enhancement.	

Table E.3: BuildingOS Assessment Approaches

**Action Item:** Review the “Approach” tab in the attached spreadsheet. Provide a comment if you believe an approach should be modified, deleted, or added.

#### E.2.2.1 Assess Player Stakeholder Experience

The goal of player assessment is to determine the effectiveness of the game framework from player’s perspective as well as the usability of the game interface and the engagement level of the game.

We recommend three approaches for player assessment: pre-post effectiveness, usability survey and engagement metrics. The attached spreadsheet outlines the planned goals and survey questionnaires for these assessment approaches, as shown in [Table E.4](#).

The first approach is **Pre-post Effectiveness Study**. One of the goals of the competition is (but not limited to) the reduction of resource such as energy and water consumption. To assess the effectiveness of this goal, we need to determine the metrics that may be measured before and after the competition (pre-post). Lucid BuildingOS and Dashboard calculates the percentage of reduction of energy and water consumption for each participated building, based on the baseline usage of the previous two weeks. We will use this metrics to measure the effect of the competition. The maximum, minimum and average percentage of reduction of all the buildings are calculated to determine the most, the least and average reduction of the resource usage. This assessment approach reveals the extend of effectiveness of the game produced by the framework, regarding to the resource consumption reduction.

1. Player Pre-Post Effectiveness Goals		
Category	Goals	Comments
Education	50% of players can name three things to save electricity	
Behavior Change	50% of players did one thing that reduces their consumption 20% of players made at least 1 conservation commitment	
Resource Reduction	3% of average reduction	

2. Player Usability Survey Questionnaire		
Question	Response type	Comments
1. What did you like most about the game?	short answer	
2. What did you find confusing?	short answer	
3. What issues did you have while using the game?	short answer	
4. What was the thing you liked the least about the game?	short answer	
5. What can we do to improve the game?	short answer	
6. It was easy to find what I was looking for on the website.	Strongly disagree - Disagree - Neutral - Agree - Strongly agree	
7. The website was responsive.	Strongly disagree - Disagree - Neutral - Agree - Strongly agree	
8. The website provided adequate help in teaching me how to play.	Strongly disagree - Disagree - Neutral - Agree - Strongly agree	
9. I understood how to play.	Strongly disagree - Disagree - Neutral - Agree - Strongly agree	
10. this is something my friends should participate in.	Strongly disagree - Disagree - Neutral - Agree - Strongly agree	

3. Player Engagement Metrics Goals		
Metric	Goals	Comments
participation (percentage of players who participated in the game)	30%	
daily player (average percentage of players per day)	5%	
daily play time (average play time of a player per day )	5 minutes	
submission (average submissions of a player)	2%	
social interaction (average social interaction of a player )	2%	
game error (percentage of players who encountered errors)	1%	

Table E.4: Player Assessment

The second approach is **Usability Survey**. We will conduct a player usability survey at the final week or right after the competition to understand the strengths and weaknesses of the game user interface perceived by players. Minimum of 20 players (the more the better) are randomly selected to participate in this survey. The survey is administrated online via survey monkey or other survey tools. We design the survey questionnaire as shown in the section 2 of [Table E.4](#). Once the survey is created online, the survey administrator will email the selected players with the link and instruction to the online survey. After we received all the survey responses, we will code and analyze the response to understand the areas of usability problems in the game interface as well as the areas of strengths. This assessment approach reveals the strengths and weaknesses of the framework regarding the usability of the game interface.

Finally, the third approach is **Engagement Metrics**. This approach calculates the engagement

metrics to assess the extent of engagement from players and the impact of the game. The more engaging the game is, the more potential impact could be to the players. We will first obtain the detailed logs of user interaction with the game. These logging includes http web server logs and user action logs which identify every user click on the web page. Once the log data are available, we will calculate the engagement metrics as described in section 3 of [Table E.4](#). With the exception of the game error metric, the higher value these metrics are, the higher engagement level the game has. Distribution of the above metrics across of the period of the competition also provides insights on the extent of engagement in different time of the competition. For example, it may be typical that the first few days of the competition may have higher number of player and play time metrics because of the launch, or due to the announcement of an interesting real-world event. This assessment reveals the extent of engagement of the players in the game.

**Action Item:** Review the Player tab in the attached spreadsheet. Provide comments for any Player assessment items that you believe might need to be changed.

### E.2.2.2 Assess System Admin Stakeholder Experience

The goal of system admin assessment is to determine to what extent the framework facilitates the system administration tasks from system admin's perspective. SGSEAM assesses how much time is required to install and maintain an instance of a serious game using the framework and the problems encountered during the system admin process.

We consider the tasks of system admin interacting with Lucid's framework are:

1. install the software
2. configure smart meter connectivity
3. backup data
4. monitor performance
5. scaling the system
6. patching

We recommend the post-hoc interview approach for system admin assessment. Once we identify the contact information of the system admins, the interview will be administrated by using an online

questionnaire form followed by an optional phone interview if needed. We design the interview with the following questionnaire that is tailored to the specific tasks of the system admins of Lucid's framework. The attached spreadsheet outlines the planned interview questionnaires, as shown in **Table E.5**.

System admin Post-hoc Interview Questionnaires	Comments on question
1. How much time did you spend to install the system and the dependencies?	
2. How much time did you spend to configure the meters?	
3. How much time did you spend to maintain the system such as backup, patching, monitoring?	
4. Did you need to scale the system? If Yes, how much time did you spend?	
5. What problems did you encounter?	
6. Did you find it difficult to admin the system? What was difficult?	
7. Can we call you for a short phone interview if we have more questions regarding your experience with the system?	

Table E.5: System Admin Assessment

Once we receive the responses from the system admin, we will code (categorize) the time and problems encountered to find out what are the problem areas if there is any. If we need further explanation to the response, we will administrate a quick phone interview to address the specific response.

These assessment reveals the strengths, weaknesses and the areas of improvement regarding the system admin process for the framework.

**Action Item:** Review the System Admin tab in the attached spreadsheet. Provide comments for any System Admin interview questionnaires that you believe might need to be changed.

#### E.2.2.3 Assess Game Designer Stakeholder Experience

The goal of SGSEAM game designer assessment is to determine the strengths and weaknesses of the framework regarding to the game design process. SGSEAM assesses how much time is required to design an instance of a serious game using the framework and the problems encountered during the design process.

We consider the tasks of game designer interacting with Lucid's framework are:

1. decide competition period
2. set up building occupancy, manual or automated meters

3. decide baseline period
4. monitor competition status during the competition

We recommend the post-hoc game designer interview approach for assessing game designer stakeholder experiences. The interview is administrated by using an online questionnaire form followed by an optional phone interview if needed. We will interview several game designers of different competitions. The more data we collect, the more insights we get. The interview is designed with the following questionnaire that is tailored to the specific tasks of the game designers of Lucid's framework. The attached spreadsheet outlines the planned interview questionnaires for game designer, as shown in [Table E.6](#).

Game Designer Post-hoc Interview Questionnaires	Comments on question
1. How much time did you spend to set up the buildings including meters?	
2. How much time did you spend to setup the competition (competition periods, baseline period, participants)?	
3. How much time did you spend to setup the homepage by deciding which widgets to include?	
4. How much time did you spend to monitor analytical data to understand the state of the game	
5. What problems did you encounter?	
6. Did you find it difficult to use the interface? What was difficult?	
7. Do you agree for us to call you for a short phone interview if we have more questions regarding your experience with the system?	

Table E.6: Game Designer Assessment

After the interview, code and categorize the reported time and problems to identify the strengths and weaknesses. In addition, if possible, collect the system log data related to the game designing tasks, analyze the logs to find out the time spent and error encountered during the game designing tasks. Use the log data to verify the findings from the interview data.

These assessment reveals the strengths, weaknesses and the areas of improvement regarding the game design process for the framework.

**Action Item:** Review the Game Designer tab in the attached spreadsheet. Provide comments for any Game Designer interview questionnaires that you believe might need to be changed.

#### E.2.2.4 Assess Game Manager Stakeholder Experience

The goal of SGSEAM game manager assessment is to determine the strengths and weakness of the framework regarding to the game management process. Similar to the assessment of the game

designer, SGSEAM assesses how much time it is required to manage an instance of a serious game using the framework and the problems encountered during the managing process.

We consider the tasks of game manager interacting with Lucid's framework are:

1. input data manually
2. manage events, marketing, handing out prizes
3. monitor competition status

We recommend the post-hoc interview approach for game manager assessment. The interview is administrated by using an online questionnaire form followed by an optional phone interview if needed. We will interview several game managers of different competitions. The more data we collect, the more insights we get. The interview is designed with the following questionnaire that is tailored to the specific tasks of the game managers of Lucid's framework. The attached spreadsheet outlines the planned interview questionnaires, as shown in [Table E.7](#).

Game Manager Post-hoc Interview Questionnaires	Comments on question
1. How much time did you spend to enter the meter data manually for the baseline period?	
2. How much time did you spend to enter the meter data manually for the competition period?	
3. What problems did you encounter?	
4. How much time did you spend to monitor analytical data to understand the state of the game	
5. Did you find it difficult to manage? What was difficult?	

Table E.7: Game Manager Assessment

After the interview, code and categorize the reported time and problems to identify the strengths and weaknesses in the game managing process. In addition, if possible, collect the system log data related to the game managing tasks, analyze the logs to find out the time spent and error encountered during the game managing tasks. Use the log data to verify the findings from the interview data.

These assessment reveals the strengths, weaknesses and the areas of improvement regarding the game managing process for the framework.

**Action Item:** Review the Game Manager tab in the attached spreadsheet. Provide comments for any Game Manager interview questionnaires that you believe might need to be changed.

### E.2.2.5 Assess Game Developer Stakeholder Experience

To investigate how easy it is to understand, extend, and debug a serious game framework from a developer's perspective, SGSEAM assesses how much time it takes to develop an enhancement to the game framework, and how many errors are encountered during the development process.

We consider the tasks of game manager interacting with Lucid's framework are:

1. use API to get data in and/or out of the system
2. customize the interface
3. extend the system to support new meters
4. enhancement

We recommend the post-hoc game developer interview approach for assessing game developer stakeholder experiences.

BuildingOS and Dashboard have APIs for developing apps to tie into the framework. We will use the API to develop an extension or customization of the system. Here are the development tasks we proposed to perform using Lucid's API to extend the framework:

1. create a new widget to be available in the home page.
2. support the automated energy data collection from a new type of meter.

We will ask the identified game developers to perform the above development tasks using Lucid's framework. The developer could be Lucid internal developers or some one outside of Lucid. After the development tasks are completed, we will interview the developers to assess his experience for these development tasks. The attached spreadsheet outlines the planned questionnaires for game developer interview, as shown in [Table E.8](#).

Game Developer Post-hoc Interview Questionnaires	Comments on question
1. How much time did you spend to implement the creation of a new widget?	
2. How much time did you spend to implement adding a new type of meter?	
3. What problem(s) did you encounter?	
4. Did you find it difficult to understand, extend and debug the system? What was difficult?	

Table E.8: Game Developer Assessment

Once the interview data is collected, categorize the reported problems and correlated with the reported time data to identify the areas of strength (less time spent) and weakness (more time spent

and problems or difficulties) in the process of development.

These assessment reveals the strengths, weaknesses and the areas of improvement regarding the game development process for the framework.

**Action Item:** Review the Game Developer tab in the attached spreadsheet. Provide comments for any Game Developer interview questionnaires that you believe might need to be changed.

### E.2.3 Choose Assessment Participants

Once the stakeholder categories are defined, the next step is to find individuals fitting those categories who will be willing to participate in the evaluation process.

Stakeholder	Person name(s)	Organization(s)	Contact(s)
Player(s)			
System admin(s)			
Game designer(s)			
Game manager(s)			
Game developer(s)			

Table E.9: Choose Participants

Table E.9 shows a sample of the *Participants* worksheet.

For each stakeholder, identify the name(s), organization and contact info. It is important to be able to contact the stakeholders in some way, either via email or phone, to get the feedback from their experiences with the framework.

**Action Item:** Review the Participants tab in the attached spreadsheet, and provide any individuals that you believe might be able to participate at this point in the planning process.

### E.2.4 Create Assessment Schedule

Once we know what the assessment approaches and who the participants are, the next step is to create the assessment schedule. We have created a sample schedule based on the sample planning timeline in the CCN Competition Planning Guide, as shown in Table E.10.

Week	CCN Milestone	CCN Task	SGSEAM Task	Comments on schedule
Mar 3 - 7		Set up buildings, meters, and competition in BuildingOS	Finalize stakeholders and assessment approaches	
Mar 10 - 14		Data collection dry run week, troubleshooting and resolve problems	Finalize interview and survey questionnaires	
Mar 17 - 21	Baseline (2 weeks)	Collect & verify baseline data	Choose participants	
Mar 24 - 28				
Mar 31 - Apr 4	Competition (3 weeks)	Kick-off, Collect competition data, Enter data into BuildingOS, manage events, Social media		
Apr 7 - 11				
Apr 14 - 18				
Apr 21 - 25		Collect, verify and enter final competition data into BuildingOS	Obtain log data, System admin post-hoc interview	
Apr 28 - May 2			Player effectiveness & usability study	
May 5 - 9			Game designer post-hoc interview	
May 12 - 16			Game manager post-hoc interview	
May 19 - 23			Game developer post-hoc interview	
May 26 - 30			Analyze data, interpret strength and weakness, produce action document	

Table E.10: Assessment Schedule

**Action Item:** Review the Schedule tab in the attached spreadsheet. Provide comments for any schedule items or dates that you believe might need to be changed.

## E.3 Step 2: Gather Data

Once the plan has been finalized, the next step is to carry out the assessment, record the data, obtain log data, and (if necessary) refine the assessment plan. The output of this step is a data repository contains all the assessment data that can be analyzed in the next step.

### E.3.1 Carry Out Assessments

For each stakeholder group, we will complete the tasks outlined in the assessment plan, gathering the data.

### E.3.2 Obtain log data

Certain assessments (such as player engagement) depend upon access to certain kinds of log data. We will confer with technical staff as to how to obtain this data.

## **E.4 Step 3: Produce Assessment Report**

In this step, we will analyze the data gathered from previous steps, create an analysis of the strengths and weakness of the framework, and produce an action report with our recommendations as to framework improvements.

### **E.4.1 Analyze Data**

Our analysis will include qualitative analysis of questionnaire data as well as quantitative analysis of log data. For example, for player assessment, we will calculate the engagement metrics from the game log; for game designer assessment, we will analyze interaction log data to find out the completion time for a certain game design tasks.

### **E.4.2 Determine Strength and Weakness**

We will attempt to determine the most important problem areas from our data and summarize them, as well as the areas where the framework appears to be most successful.

### **E.4.3 Produce Report with Actionable Steps**

Once the strengths and weaknesses of the framework are identified from the data analysis, an action report should be produced. This report includes the weakness areas that can be improved and actionable steps on how to improve from each stakeholder's perspective. It also includes strengths that the framework needs to maintain.

This concludes the SGSEAM assessment for Lucid BuildingOS and BuildingDashboard.

## **APPENDIX F**

### **QUESTIONNAIRE FOR SGSEAM IN-GAME SURVEY**

This appendix details the questionnaire of the in-game surveys as part of the SGSEAM assessment for the player stakeholder experiences. The in-game surveys were made available to players as an action in the Smartgrid game during the UHM Kukui Cup challenges. The surveys were hosted in the SurveyGizmo [?] website. The survey started with the informed consent via inputting the email address.

The questions were prefaced with the following instructions: “The goal of this survey is to learn about your experiences during the Kukui Cup. Please answer honestly, we want to know how you really feel.”

This appendix only lists the questions that are related to the Makahiki framework.

#### **F.1 UHM 2011 in-game survey questionnaire**

1. Prior to playing the Kukui Cup, were you interested in energy conservation?
2. Has the Kukui Cup increased your interest in energy conservation and sustainability?
3. Have you made any commitments through the website during the game?

Yes

No

[If Yes] Did you change your behavior during the competition based on the commitment(s) you made?

Yes

No

Not sure

4. How much time do you usually spend on the following activities? [Options for each activity:]

1. 3 or more hours a day

2. about 1 hour a day
3. about 1 hour a week
4. 1 hour a month or less
5. never

List of activities:

- Playing games on a laptop computer
- Playing games on a game console (Xbox, PS3, Wii)
- Playing games on a handheld game device (DS3, PSP)
- Playing games on a mobile phone
- Checking Facebook
- Checking Twitter

5. Rate how much you agree with each statement below: [Each statement was ranked on the following Likert-type scale:]

1. Strongly disagree
2. Disagree
3. Neutral
4. Agree
5. Strongly agree
6. Not Applicable

Statements:

- It was easy to find what I was looking for in the website.

- The website was responsive. I did not wait too long after I clicked on something.
- The website provided adequate help in teaching me how to play the game.
- I understood the rules of the game and how to play.

6. What did you like about the website?

7. What did you find confusing about the website?

8. If you could add or change something in the website, what would that be?

9. How would you describe the Kukui Cup? (check all that apply)

Fun

Educational

So-so

Boring

Not useful

Difficult

Addictive

Other: \_\_\_\_\_

10. The Kukui Cup website shows energy data updated every 15 seconds. Did you find this helpful in conserving energy?

not really, updating the data daily would be enough

not really, updating the data hourly would be enough

not really, I only care about the final result of the competition

yes, it is helpful to see the energy usage changing in real time

11. Which of the following do you wish there were more of in the game? (choose all that apply)

- events
- excursions
- commitments
- videos
- social activities
- physical activities
- online activities

12. On average, how many minutes a day did you spend on the Kukui Cup website?

13. If you were able to play the Kukui Cup next year, would you?

- Yes
- I enjoyed it, but I wouldn't play again
- I didn't enjoy it, and I wouldn't play again
- No, because: \_\_\_\_\_

14. How likely would you be to recommend playing the Kukui Cup to a first year student in Fall 2012?

- Very Likely
- Likely
- Neutral
- Unlikely
- Very Unlikely
- Not Applicable

15. Is there anything else you would like to tell us about your experience playing the Kukui Cup that this survey didn't ask?

## F.2 UHM 2012 in-game survey questionnaire

1. Prior to playing the Kukui Cup, were you interested in energy conservation? Please answer with a number in a Likert scale from 1 to 7, where 1 is “Very Interested” and 7 is “Not at all Interested”.

Very Interested    1    2    3    4    5    6    7    Not at all Interested

2. Has the Kukui Cup increased your interest in energy conservation and sustainability? If so, how?

3. Did you change your behavior during the competition based on the commitment(s) you made? If so, how?

4. The Kukui Cup website shows real-time energy data for your team. Did you find this helpful in conserving energy?

5. Do you find the Kukui Cup website responsive? In other words, did pages load quickly?

Extremely quickly

Quickly

OK

Slow

Very slow

6. Was it easy to understand how to unlock new activities and levels?

Yes

No

7. What was the thing you liked most about the Kukui Cup so far?

8. What was the thing you liked the least about the Kukui Cup so far?

9. If you were able to play the Kukui Cup next year, would you? Please answer with a number in a Likert scale from 1 to 7, where 1 is “Absolutely I would” and 7 is “Definitely Not”.

Absolutely I would    1    2    3    4    5    6    7    Definitely Not

10. Did you try accessing the application from a smart phone? If so, please tell us what kind of phone you used and whether or not you encountered any problems.
11. How likely would you be to recommend playing the Kukui Cup to a first year student in Fall 2013? Please answer with a number in a Likert scale from 1 to 7, where 1 is “Likely” and 7 is “Unlikely”.

Likely    1    2    3    4    5    6    7    Unlikely

12. What did you find confusing about the website (if anything)?
13. What do you consider to be the single most important thing for us to improve in the website?
14. Is there anything else you would like to tell us about your experience playing the Kukui Cup that this survey didn’t ask?

### **F.3 UHM 2014 in-game survey questionnaire**

1. It was easy to find what I was looking for in the website.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree
- Not Applicable

2. The website was responsive. In other words, did pages load quickly?

- Strongly disagree
- Disagree
- Neutral
- Agree

Strongly agree

Not Applicable

3. The website provided adequate help in teaching me how to play.

Strongly disagree

Disagree

Neutral

Agree

Strongly agree

Not Applicable

4. If you were able to play the Kukui Cup next year, would you?

Strongly disagree

Disagree

Neutral

Agree

Strongly agree

Not Applicable

5. This is something my friends should participate in.

Strongly disagree

Disagree

Neutral

Agree

Strongly agree

Not Applicable

6. I understood how to play

Strongly disagree

Disagree

Neutral

Agree

Strongly agree

Not Applicable

7. What did you like the most about the game?

8. What did you found confusing about the website?

9. What issues did you have while using the game?

10. What was the thing you liked the least about the game?

11. What can we do to improve the game?

12. When you logged in for the first time, you are prompted with a series of screens to set up your profile. What did you think of this "first login sequence"?

13. After finishing the first login sequence, the application took you to the home page. Did you know what to do next?

14. Is there anything else you would like to tell us about your experience playing the Kukui Cup that this survey didn't ask?

## **APPENDIX G**

### **GOOGLE FORMS FOR SGSEAM IN-LAB EXPERIMENTS**

This appendix lists the Google forms that were used in the SGSEAM in-lab experiment approach for system admin and game designer assessments. They were presented to the students in the ICS 691 class at the University of Hawaii at Manoa in Spring 2013 as parts of their class assignments. The students were voluntarily participated in these assessment experiments.

#### **G.1 System admin Assessment**

There are two google forms to assess the system admin's experience with the Makahiki framework regarding the installation of Makahiki and its software dependencies, based on Makahiki documentation provided online. One google form is the Local installation log to assess the local installation of Makahiki on their personal computers or on a virtual machine created on their personal computers, the other is the Heroku installation log to assess the cloud installation of Makahiki on the Heroku environment.

##### **G.1.1 Makahiki Local Installation Google Form**

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

##### **2.1.1.1.2. Install Python \***

Complete the “Install Python” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-python>,

record the time you spent for this section only:

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

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

#### **2.1.1.1.3. Install C Compiler \***

Complete the “Install C Compiler” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-c-compiler>,

record the time you spent for this section only:

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

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

#### **2.1.1.1.4. Install Git \***

Complete the “Install Git” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-git>,

record the time you spent for this section only:

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

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

#### **2.1.1.1.5. Install Pip \***

Complete the “Install Pip” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-pip>,

record the time you spent for this section only:

- 0 minute (Already installed from previous assignments)
- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

#### **2.1.1.1.6. Install Virtual Environment Wrapper \***

Complete the “Install Virtual Environment Wrapper” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-virtual-environment-wrapper>,

record the time you spent for this section only:

- 0 minute (Already installed from previous assignments)
- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record the problem you encountered when installing virtual environment wrapper:

---

#### **2.1.1.1.7. Install Python Imaging Library \***

Complete the “Install Python Imaging Library” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-python-imaging-library>, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

---

#### **2.1.1.1.8. Install PostgreSQL \***

Complete the “Install PostgreSQL” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-postgresql>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

#### **2.1.1.1.9. Install Memcache \***

Complete the “Install Memcache” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-memcache>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

#### **2.1.1.1.10. Download the Makahiki source \***

Complete the “Download Makahiki source” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#download-the-makahiki-source>, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record the problem you encountered when download the Makahiki source: \_\_\_\_\_

#### **2.1.1.1.11. Workon Makahiki \***

Complete the “Workon Makahiki” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#workon-makahiki>,

record the time you spent for this section only::

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

---

#### **2.1.1.1.12. Install required packages \***

Complete the “Install required packages” section in Makahiki Local Installation Manual:

[http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-required-](http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#install-required-packages)

packages, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

#### **2.1.1.1.13. Setup environment variables \***

Complete the “Setup environment variables” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#setup-environment-variables>, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record the problem you encountered when setting up environment variables: \_\_\_\_\_

#### **2.1.1.1.14. Initialize Makahiki \***

Complete the “Initialize Makahiki” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#initialize-makahiki>, record the time you spent for this section only:

- 5 minutes

- 10 minutes
- 30 minutes
- 1+ hour

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

#### **2.1.1.1.15. Start the server \***

Complete the “Start the server” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#start-the-server>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when starting the server: \_\_\_\_\_

#### **2.1.1.1.16. Verify that Makahiki is running \***

Complete the “Verify that Makahiki is running” section in Makahiki Local Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-unix.html#verify-that-makahiki-is-running>, record the time you spent for this section only:

- 5 minutes
- 10 minutes

- 30 minutes
- 1+ hour

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

Your UH email: \* \_\_\_\_\_

### **G.1.2 Makahiki Heroku Installation Log**

Please follow the steps outlined in this form to install Makahiki on Heroku 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

#### **2.1.1.2.1. Install Heroku \***

Complete the “Install Heroku” section in Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#install-heroku>,

record the time you spent for this section only:

- 0 minute (Already installed from previous assignments)
- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered when installing Heroku: \_\_\_\_\_

#### **2.1.1.2.2. Add your SSH keys to Heroku \***

Complete the “Add your SSH keys to Heroku” section in Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#add-your-ssh-keys-to-heroku>, record the time you spent for this section only:

- 0 minute (Already installed from previous assignments)
- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when adding your SSH keys to Heroku: \_\_\_\_\_

#### **2.1.1.2.3. Verifying your Heroku account \***

Complete the “Verifying your Heroku account” section in Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#verifying-your-heroku-account>, record the time you spent for this section only:

- 0 minute (Already installed from previous assignments)
- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when verifying your Heroku account: \_\_\_\_\_

#### **2.1.1.2.4. Setup Amazon S3 \***

Complete the “Setup Amazon S3” section in Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#setup-amazon-s3>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when setting up S3: \_\_\_\_\_

#### **2.1.1.2.5. Setup environment variables \***

Complete the “Setup environment variables” section in Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#setup-environment-variables>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when setting up environment variables: \_\_\_\_\_

#### **2.1.1.2.6. Download the Makahiki source \***

Complete the “Download the Makahiki source” section in the Makahiki Heroku Installation Manual:

[http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#download-the-](http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#download-the-makahiki-source)

makahiki-source, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when download the Makahiki source: \_\_\_\_\_

#### **2.1.1.2.7. Initialize Makahiki \***

Complete the “Initialize Makahiki” section in the Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#initialize-makahiki>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when initializing Makahiki: \_\_\_\_\_

#### **2.1.1.2.8. Start the server \***

Complete the “Start the server” section in the Makahiki Heroku Installation Manual:

<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#start-the-server>,

record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when starting the server: \_\_\_\_\_

#### **2.1.1.2.9. Verify that Makahiki is running \***

Complete the “Verify Makahiki is running” section in the Makahiki Heroku Installation Manual:

[http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#verify-that-makahiki-](http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#verify-that-makahiki-is-running)

is-running, record the time you spent for this section only:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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

Your UH email: \* \_\_\_\_\_

## **G.2 Game designer Assessment**

There is one google form to assess the game designer's experience with the Makahiki framework regarding the game design process using Makahiki, based on Makahiki documentation provided online.

### **G.2.1 Makahiki Configuration Log Google Form**

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
- 10 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-design-page>.

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

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes

- 1+ hour

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:

- 5 minutes
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes

- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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 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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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
- 10 minutes
- 30 minutes

- 1+ hour

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
- 10 minutes
- 30 minutes
- 1+ hour

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

\_\_\_\_\_

Your UH email: \* \_\_\_\_\_