www.itcon.org - Journal of Information Technology in Construction - ISSN 1874-4753

DEVELOPMENT OF ECOCAMPUS: A PROTOTYPE SYSTEM FOR SUSTAINABLE BUILDING DESIGN EDUCATION

SUBMITTED: September 2014
REVISED: December 2014
PUBLISHED: December 2014 at http://www.itcon.org/2014/30
EDITOR: Rezgui Y.

Steven K. Ayer, Assistant Professor
Arizona State University
steven.ayer@asu.edu, http://engineering.asu.edu/sebe

John I. Messner, Charles and Elinor Matts Professor of Architectural Engineering
The Pennsylvania State University
jmessner@engr.psu.edu, http://www.engr.psu.edu/ae/cic/

Chimay J. Anumba, Department Head and Professor of Architectural Engineering
The Pennsylvania State University
anumba@engr.psu.edu, http://www.engr.psu.edu/ae/faculty/anumba/index.asp

SUMMARY: *Students in architectural engineering and related disciplines need to develop critical design skills to be successful in their post-collegiate careers. Among these critical skills are the abilities to: create possible design concepts to address a given functional or programming requirement; understand what those concepts will look like in the context of an existing building or project site; and assess the benefits and deficiencies of the design concepts to determine the best possible design strategy. Existing technologies have been used to help students develop these skills through a mobile computer-based prototype, called ecoCampus. This computer application allowed students to: explore sustainable retrofit design possibilities for an existing building; visualize those design possibilities with marker-based augmented reality; and receive tailored performance feedback about the designs through a basic simulation game interface. This paper discusses the general workflow for creating this type of tool and the specific steps that were followed to develop ecoCampus. These steps include: identifying educational objectives; developing a list of functional requirements; developing the workflow for the educational design module; determining the development infrastructure for creating the application; developing the tool; and refining the initial prototype through iteration. This workflow is offered as a process that can successfully lead to a functional prototype as evidenced by the creation of ecoCampus, and can be used by future developers for creating similar technological tools applied to various educational disciplines.*

KEYWORDS: *ecoCampus, simulation game, augmented reality, engineering education*

REFERENCE: *Steven K. Ayer, John I. Messner, Chimay J. Anumba (2014). Development of ecoCampus: a prototype system for sustainable building design education, Journal of Information Technology in Construction (ITcon), Vol. 19, pg. 520-533, http://www.itcon.org/2014/30*

1. INTRODUCTION

The process of developing an effective building design requires architects and engineers to consider several possible design options. These different design concepts can be analyzed to determine the best option for a given building project with particular performance needs. Traditionally, architects and engineers have illustrated their chosen design concepts through 2D paper design documents. For new engineers, the process of creating a mental model from the 2D documents can be challenging and prone to errors (Johnson, 1997). Despite the challenges associated with visualizing building design content with these traditional methods and the importance in overcoming these challenges, visualization is seldom stressed in education (Finke, 1990; Finke et al., 1996; Rieber, 1994). This research examines how technology can be leveraged to aid engineering students in visualizing and assessing potential building design concepts.

A prototype educational software application called ecoCampus was created to help students visualize design modifications to a building component of an existing facility and receive tailored performance feedback about their design decisions to assess performance. For the developed prototype, the primary design goal was to improve the sustainable performance of an exterior curtain wall of an existing building on a college campus. This ecoCampus prototype helped students to visualize designs more easily than paper-based methods and allowed them to complete a more thorough design process through the development of more design options (Ayer et al., 2013a; b; c).

This paper presents related prior studies, as well as the process that was followed for the creation of the ecoCampus prototype. The specific development steps that were followed are presented and the development pitfalls that were observed are also discussed. Additionally, the specific development steps were generalized so that future researchers may apply the software development methodology employed in this work to create similar educational games for other educational topics in the future.

2. BACKGROUND

To help users visualize building designs, ecoCampus used an augmented reality interface to allow students to see designs in the context of the existing building space. Augmented Reality (AR) technology helps to blur the line between real and virtual worlds by superimposing computer generated imagery onto a user's, predominantly real view of a space (Milgram and Kishino, 1994). A common example of this type of technology would be the yellow 'first down line' that appears on most televised American football games. When targeted to design applications, this technology can help provide a more intuitive method for students to visualize problem content (Azuma, 1997; Furmanski et al., 2002).

In addition to the visualization benefits of AR, ecoCampus also sought to help students to assess the performance of different building design concepts with a basic simulation game interface. Simulation games are defined as contests where users move toward specific goals under sets of conditions and constraints that will sufficiently model a real-world situation (Gredler, 1994; Jacobs and Dempsey, 1993). This type of environment can be beneficial as a learning tool because it frames learning content in a specific situation (Aldrich, 2004, 2005; Bartle, 2003; Brown et al., 1989). Situated learning theory suggests that the best way for students to learn content that will eventually be applied to a given context is for them to learn the content within the context in which it will be applied (Lave and Wenger, 1991). In the case of this work, ecoCampus helps to remove the common barriers that are present for new engineering design students to visualize and assess different design options. This allows the students to move past these challenges and actually consider different design options so that they may learn sustainable principles and good design practices through the completion of an actual building design activity.

2.1 Software Development Frameworks

A variety of software development frameworks have been identified for different types of development efforts. For development tasks where software requirements are very well defined, conventional development approaches, such as the waterfall model may be effective (Royce, 1970). This approach encourages fewer prototype iterations to be developed, but more focus on the effort involved in each sequential step involved in creating the application. This approach was not as well suited for this work because of the uncertainty in definition of the developed application from the start of the project. For applications where there is minimal specification for an application or where the final performance requirements may be frequently changing, prior works suggest benefits of using Rapid Application Development (Gottesdiener, 1995) or Extreme Programming (Beck, 1999) approaches. These approaches stress putting effort into actual software development and allowing knowledge to be gained during testing to direct subsequent development efforts. These approaches also did not align with the needs of the ecoCampus development effort because, while the exact specifications of the applications were not immediately defined, the overall educational intent of the program was well defined from the start.

For this development work, Iterative and Incremental Development frameworks were more closely aligned with the development needs (Larman, 2003). Iterative and Incremental Development frameworks such as the spiral model (Boehm, 1988) and prototyping models (Luqi, 1989) were best suited for this work because these models encourage analysis and design of a computer application, but also suggest development of multiple iterations of the application in various levels of development to solicit additional formative feedback about performance. This type of development approach allows for appropriate application definition to allow for the guiding educational principles to remain constant, but also allows for additional specifications to be incorporated related to applications functions that were not foreseen in the initial design stages.

The prior works defining software development frameworks are helpful in determining a high-level strategy for approaching software development. As a result, they are very generalizable and can be applied to many different types of software development efforts. The work presented in this paper presents a more directed software development methodology intended to offer a structured workflow for future developers to create mixed reality-based educational games in a variety of disciplines.

3. METHODOLOGY

This section describes the process followed to create the ecoCampus prototype. The process described is not suggested to be the only method for creating an augmented reality-based simulation game. Instead, the methodology described illustrates a structured approach that can successfully create a functional augmented reality-based simulation game application as validated by the successful creation of ecoCampus. The specific steps used to create ecoCampus are listed in the top part of Figure 1. These specific steps were also generalized and shown in the bottom portion of Figure 1 to help developers apply the process used for ecoCampus to future development efforts. Each of these development processes are described in detail in the following sections.

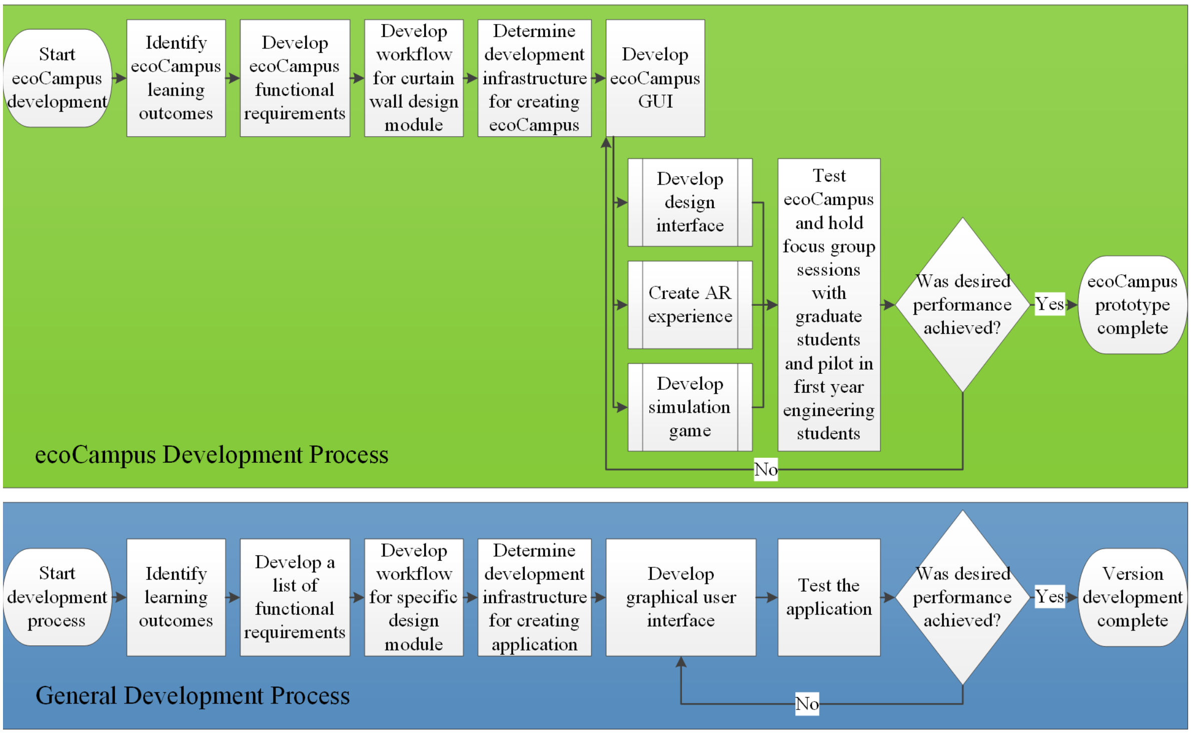

Start ecoCampus development
Identify ecoCampus leaning outcomes
Develop ecoCampus functional requirements
Develop workflow for curtain wall design module
Determine development infrastructure for creating ecoCampus
Develop ecoCampus GUI
Develop design interface
Create AR experience
Develop simulation game
Test ecoCampus and hold focus group sessions with graduate students and pilot in first year engineering students
Was desired performance achieved?
Yes
ecoCampus prototype complete
No
ecoCampus Development Process
Start development process
Identify learning outcomes
Develop a list of functional requirements
Develop workflow for specific design module
Determine development infrastructure for creating application
Develop graphical user interface
Test the application
Was desired performance achieved?
Yes
Version development complete
No
General Development Process

Figure 1: ecoCampus development process map (top) along with a generalized development process (bottom).

3.1 Identify learning outcomes

Tools are created to allow users to address needs or problems that they could not solve as effectively without the tool. In an educational context, the need of the user is to learn certain content. To create an effective educational tool, the first key step is to clearly identify the learning objectives or learning outcomes that the new tool will facilitate.

In developing the ecoCampus experience, this process of creating specific educational objectives helped to direct development efforts. The objectives developed related to getting students to consider and recognize different sustainability principles related to the design task as well as to brainstorm different possible design solutions to determine a best solution to the challenge provided. The following learning outcomes were targeted for the developed prototype:

- Design a new exterior wall to improve sustainability in an existing building;
- Brainstorm several possible design ideas in the design process;
- Consider possible building materials other than just those used in the existing design;
- Consider the visual implications of a wall design in the context of the surrounding space;
- Recognize examples of building design tradeoffs related to curtain wall redesign;
- Critically assess performance benefits and drawbacks of designs;
- Critically assess performance benefits and drawbacks of different building materials; and
- Recognize the differences between initial costs and lifecycle costs for different designs.

3.2 DEVELOP LIST OF FUNCTIONAL REQUIREMENTS

While the defined educational objectives reflect the desired learning tasks for the students, the functional requirements reflect the desired performance outcomes for the computer application. This set of functional requirements can help guide the development effort to allow users to achieve the educational objectives. Additionally, this list of functional requirements also helps when determining the appropriate development infrastructure.

For the development of ecoCampus, the high level functional requirements developed were that the tool would need to:

- Be easy to use;
- Allow a user to define a building design;
- Allow a user to visualize design content in AR; and
- Calculate basic performance values to provide a user with simulated feedback.

3.3 DEVELOP WORKFLOW FOR EDUCATIONAL MODULE

After the high level functional requirements were developed for the application, a more specific workflow was devised to specify how a user would interact with the tool to achieve the defined educational objectives. This workflow was illustrated through storyboarding. The process of creating these storyboards was an iterative one, which required brainstorming of workflow options and then analysis of different options to create a functional workflow. This process also helped to conceptualize a graphical user interface (GUI) for the developed tool.

For ecoCampus, creating storyboard images required specific consideration of both the targeted educational objectives as well as the functional requirements of the application to determine an effective user experience. Through iterations, the specific ecoCampus workflow evolved. Several of the initial storyboards are shown in Figure 2. Through analysis of these initial storyboard images, it was determined that they were more cumbersome and complicated than they needed to be. As different possible GUI options were discussed, the storyboard images were modified to reflect an application workflow that would allow users to achieve the targeted educational objectives with minimal unnecessary complexity.

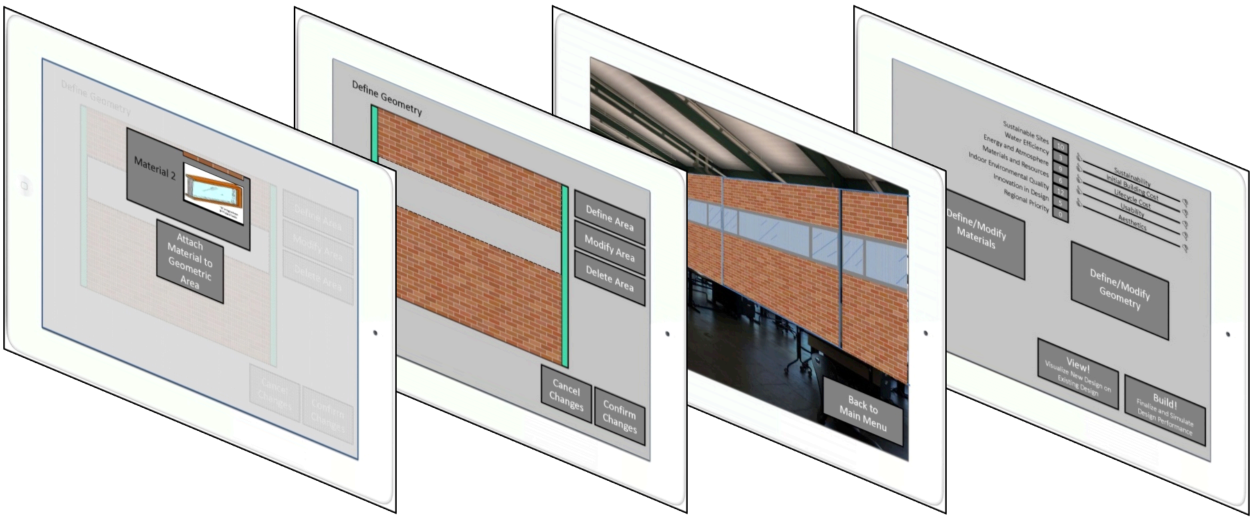


Figure 2: Initial storyboard images were developed to help define the ecoCampus workflow.

3.4 DETERMINE DEVELOPMENT INFRASTRUCTURE

After a GUI workflow was planned for the application, the necessary development infrastructure needed to be determined. This process involved identifying possible game development tools, and mobile computing device platforms. For each option, the comparative benefits were identified. It was important to not only consider the benefits of different individual components, but also to consider whether the different components would be compatible for creating the application.

For the development of ecoCampus, there were several key considerations when determining an appropriate development infrastructure. To realize a marker-based AR interface as planned, the application needed to be created for a lightweight mobile computing platform that would allow devices with a camera to track a printed fiducial marker. The computing device would also need to have the graphical capability to display 3D model content and generate simulated performance values for a particular design based on predefined algorithms. The two computing operating systems that were initially identified as options for this work were Apple iOS® and Google Android™. Both operating systems allowed for lightweight mobile devices with cameras to be able to handle the built application. From a development perspective, it was of particular interest to create ecoCampus in such a way that the developed content could be easily exported to either of these mobile platforms.

For development of game content, Unity® was identified as an option. Unity is a game engine that is currently available at no cost for creating video games for a variety of game system formats including, Apple iOS®, Google Android™, Xbox 360®, PlayStation 3®, Nintendo Wii®, and web and PC-executable formats. Unity allows developers to create 3D virtual worlds and create graphical user interfaces with which users can interact. With the addition of some optional license upgrades to Unity, game content can be exported to different game system formats. This was seen as an important benefit because it meant that once ecoCampus content was created in Unity, with only minimal modifications, it could be sent to other game system formats.

In addition to the developmental flexibility that was offered for exporting created game content through the use of Unity, many other game developers have created plug-ins to Unity to facilitate certain game functionality. For this work, String® (www.poweredbystring.com) was identified as a beneficial plug-in for Unity. String is used to help facilitate marker based AR. The plug-in includes some scripts that allow the game camera in Unity to be linked to the camera on a mobile computing device and search for a predefined printed fiducial marker. When a marker is identified, 3D game content can be displayed in a defined position relative to the printed marker.

After the components in the development infrastructure were determined, the compatibility of all components was considered. For the defined ecoCampus infrastructure, there were many necessary components that would need to work together to allow for ecoCampus to be successfully built to a mobile computing device. For initial ecoCampus development efforts using Apple's iOS platform, ecoCampus used the String plug-in to Unity, Unity, and Apple's Xcode® compiler on a computer running Apple OS X® to eventually build to Apple iOS. From an interoperability standpoint, this series of components created the potential for challenges. From preliminary exploration of this workflow, it was observed that if any one of these different components was not compatible with the rest, the entire game development build would fail. Therefore, when compatible versions were installed, a point was made to not upgrade any of the components to newer versions to avoid breaking the chain of compatibility. The specific components and versions used for development of ecoCampus can be seen in Table 1.

Table 1: ecoCampus development components and specifications

Necessary component	**Chosen product**	**Component Specifications**
Development Computer	Apple iMac	• 2.7GHz Intel Core i5 processor • 4GB 1333MHz DDR3 Memory • AMD Radeon HD 6770M 512MB Graphics Card • Mac OSX Lion 10.7.4
Mobile computing platform	Apple iPad (3rd Generation)	• iOS Version: 6.0
Application compiler	Xcode	• Version 3.5
Graphical user interface (GUI) development environment	Unity Game Engine	• Version 3.5.4
Marker-based augmented reality technology	String	• Developer License Version 1-1-3

3.5 CREATE APPLICATION

After a workable development infrastructure was determined, the actual game development process could begin. This iterative process required developers to create the core functionality and graphical user interfaces that were defined during the educational module workflow development phase. Depending on the nature of a particular game development effort, some of the planned educational workflows could potentially need to be modified slightly for development. If this happens, it is advisable to re-examine the new workflow to ensure that it still meets all educational goals and objectives. The following sections describe the specific workflow that was followed for the creation of ecoCampus.

3.6 DEVELOP GRAPHICAL USER INTERFACE

For ecoCampus, the different user interfaces were developed as different scenes within Unity and GUI elements, such as buttons, information windows, and slider bars were incorporated into the game according the planned educational module workflow. Ultimately, the ecoCampus GUI included three primary scenes with which a user would interact. These consisted of one where a user would design a new exterior wall for the existing building used in this work, another scene where a user would view design content superimposed on top of the existing building design with AR, and a third scene where a user would receive tailored simulation feedback about a design. Each different scene incorporated into ecoCampus allowed for unique game objects, buttons, and camera behaviors to be developed. This meant that for the design and simulation scenes, a user would not have control of the game camera position, which would provide a 2D virtual experience. For the augmented reality scene, the camera would switch from a fixed position camera to a live view from the camera on the mobile computing device. This allowed users to switch from designing in 2D to viewing navigable 3D AR versions of their designs.

One of the challenges observed while creating this AR interface in ecoCampus was getting virtual content to be scaled properly with the real content based on the printed fiducial marker. String would display 3D virtual content based on the size and position of the printed marker. Therefore, before virtual content could be scaled, it was important to determine the appropriate size for a fiducial marker for tracking purposes. Different marker sizes were considered. In attempting to determine an appropriate marker size, it was observed that there was a tradeoff between larger, stable markers and smaller markers that make it easier for a user to move around a space while still keeping the entirety of the marker in view. Figure 3 shows the two different marker sizes that were considered during this work. A 60"x80" (1.52m x 2.03m) marker on the left was initially explored, but because of the large size made it difficult for users to get very close to the wall to explore the augmented content in detail. The second 40"x55" (1.01m x 1.40m) marker on the right proved to be a better solution as it still allowed for stable tracking from a distance, but also allowed users to keep the marker in their view and maintain their augmented view as they would approach the existing wall as well.

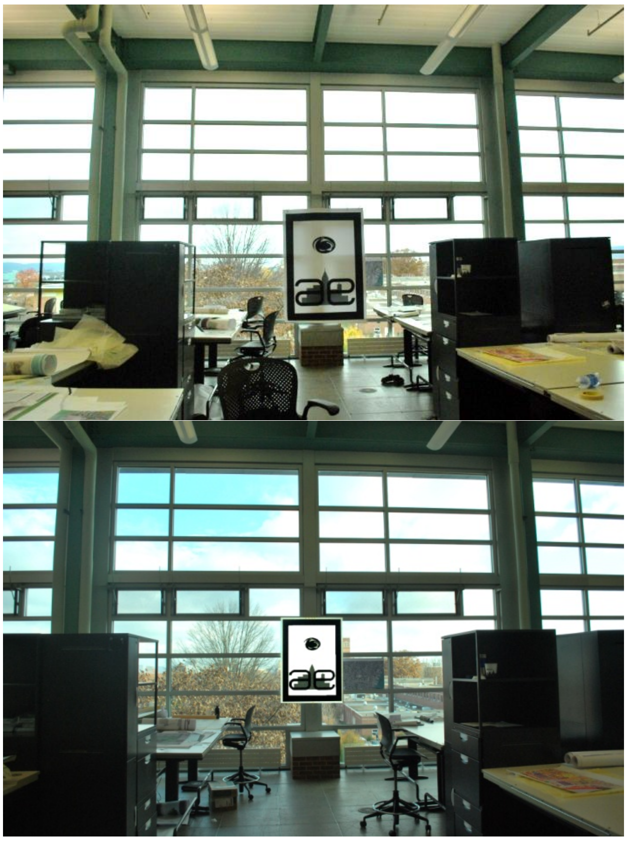
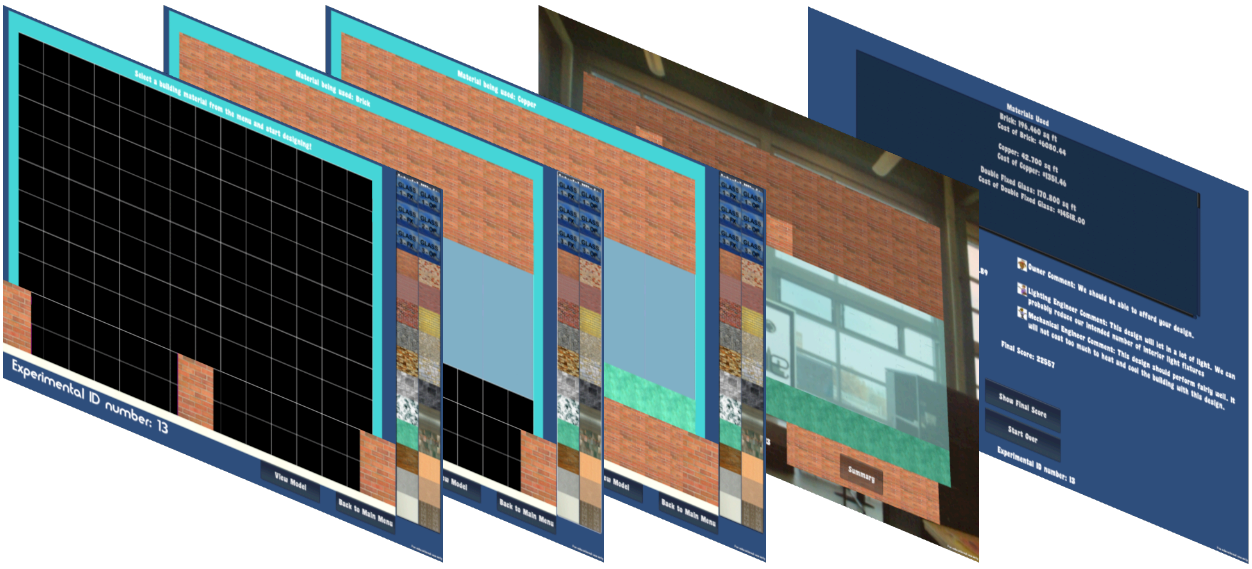


Figure 3: Different fiducial marker sizes were explored to determine the most appropriate marker size.

After an appropriate marker size was determined, a scaled mockup of the actual curtain wall with the actual fiducial marker hung on the wall was created in the computer lab where ecoCampus was developed. There was not sufficient space to recreate the entire curtain wall in the lab, so the brick pier that can be seen below the printed marker in Figure 3 was replicated by taping pieces of paper to a wall of the computer lab with the actual marker hung at the appropriate distance above the brick pier. This simple mockup provided a reference to help to properly scale the virtual content.

After the scaling issue was resolved, the ecoCampus game workflow was developed and tested in the computer lab on an iPad with the paper mockup as a reference for augmented content. The three main user interfaces allowed users to design, visualize, and receive feedback on their wall concept. The design interface allowed users to see a virtual representation of the existing wall with a blank area where they could select materials from the library and apply the selected materials to predefined wall geometry. In the AR scene, the previously defined building materials were rendered to the corresponding wall geometry and displayed when the fiducial marker was identified. The third interface allowed users to view simulation performance and game results. This performance content was generated by running count functions to determine the quantities of different building materials used, and subsequently computing several performance metrics from those quantities. The basic ecoCampus workflow used can be seen in Figure 4.


Select a building material from the menu and start designing!

Material being used: Brick

Material being used: Copper

Materials Used
Brick: 196.460 sq ft
Cost of Brick: $6080.44
Copper: 42.700 sq ft
Cost of Copper: $1351.46
Double Fixed Glass: 170.800 sq ft
Cost of Double Fixed Glass: $14518.00

Owner Comment: We should be able to afford your design.

Lighting Engineer Comment: This design will let in a lot of light. We can probably reduce our intended number of interior light fixtures

Mechanical Engineer Comment: This design should perform fairly well. It will not cost too much to heat and cool the building with this design.

Final Score: 22557

Experimental ID number: 13

Figure 4: ecoCampus design workflow

3.7 PERFORM APPLICATION ALPHA TESTING

When the initial ecoCampus prototype was developed according to the defined educational module workflow, it was tested with graduate students to determine any critical usability issues. This test identified a major issue with the workflow. Initially, the application was tested in the existing building during the day when the sun was shining through the exterior wall. This backlighting of the fiducial marker forced the iPad® camera to dim to adjust for all of the daylight and caused a loss of the fiducial marker from focus. This issue was addressed by having students use ecoCampus in the evening hours after the sun had set, but that also introduced a limitation to the application.

In further tests with graduate students during the evening hours, other bugs and workflow issues were identified. For example, some students mentioned that in the performance summary user interface, all of the simulation content was displayed immediately upon entering the scene. This caused some users to feel a sense of information overload in trying to digest all of the simulation content at once. It was suggested to animate the text so that it would take time to float onto the screen and help to direct a user's eye to each piece of simulation feedback.

As these tests were completed with graduate students and suggestions were generated for improving the ecoCampus workflow, the feedback was organized and prioritized. The most critical feedback related to issues that would limit the functionality of ecoCampus was directly incorporated into the developed prototype. Other feedback was saved for future development iterations of ecoCampus, and not directly incorporated into the first version of the application.

3.7.1 Perform Application Beta Testing with First-Year Students

The developed prototype was tested with 47 students in a first-year seminar course in the fall 2012 semester. In addition to the pedagogical findings from this implementation, several development-related suggestions were also developed from the participating students related to how the ecoCampus prototype could be improved for future semesters. From assessment tests and focus group sessions, it was observed that students wanted the ability to:

1. Develop design concepts faster in the design interface;
2. Have more choices for different materials to use while designing;
3. Distinguish between different glass materials while designing; and
4. Create more realistic design feedback from the virtual critics.

These suggestions were explored in detail to determine how to modify the ecoCampus prototype to address the feedback.

3.7.2 Modify Graphical User Interface

The first suggestion, related to speeding up the design process using ecoCampus was addressed by changing the way that users would interact with the GUI buttons created in Unity. In the first ecoCampus prototype, users would touch a material from the material library to select an active material and then individually touch each piece of curtain wall geometry to apply the active material. With the curtain wall bay subdivided into 140 different pieces, individually selecting each piece during design became time consuming. For the second version of ecoCampus, the user interaction with the curtain wall geometry was modified to use a ray cast method for registering design selections. In this second version of ecoCampus, when a user would touch and drag a finger across the curtain wall geometry, any geometry that would collide with the swipe path would have the active material assigned to it. This effectively allowed a user to select an active material and swipe their finger across the screen to quickly apply a material to several pieces of geometry. This greatly reduced the time that was necessary to create a curtain wall design concept.

Future development efforts could also explore alternate approaches to facilitating more efficient design generation. The ecoCampus GUI could be modified so that it would not require users to apply materials to predetermined wall geometry, but instead could allow them to define custom geometry for all of their material selections. A change such as this could have a variety of impacts on the use and learning outcomes. It could speed up the design process by allowing users to define large sections of wall geometry in one finger swipe and it could also encourage design creativity with the increased design freedom. However, a change like this could also potentially slow students by requiring them to think more about the form characteristics of their concept while they design. The "right" development approach for a particular application may vary based on the targeted learning outcomes. For the development of ecoCampus, the learning outcomes were largely focused on building performance, so the predefined wall geometry with the incorporation of a finger swipe function was determined to be an acceptable solution.

The second suggestion that was identified during the fall 2012 ecoCampus implementation was related to the number of materials that were incorporated into the first ecoCampus version. In the first version of the tool, students were given 28 materials with which they could design. The second version of ecoCampus allowed students to have 41 different building material options to use during design. This addition of materials required the design interface's material library to be modified. Figure 5 shows the material library layout for the first and second versions of ecoCampus.

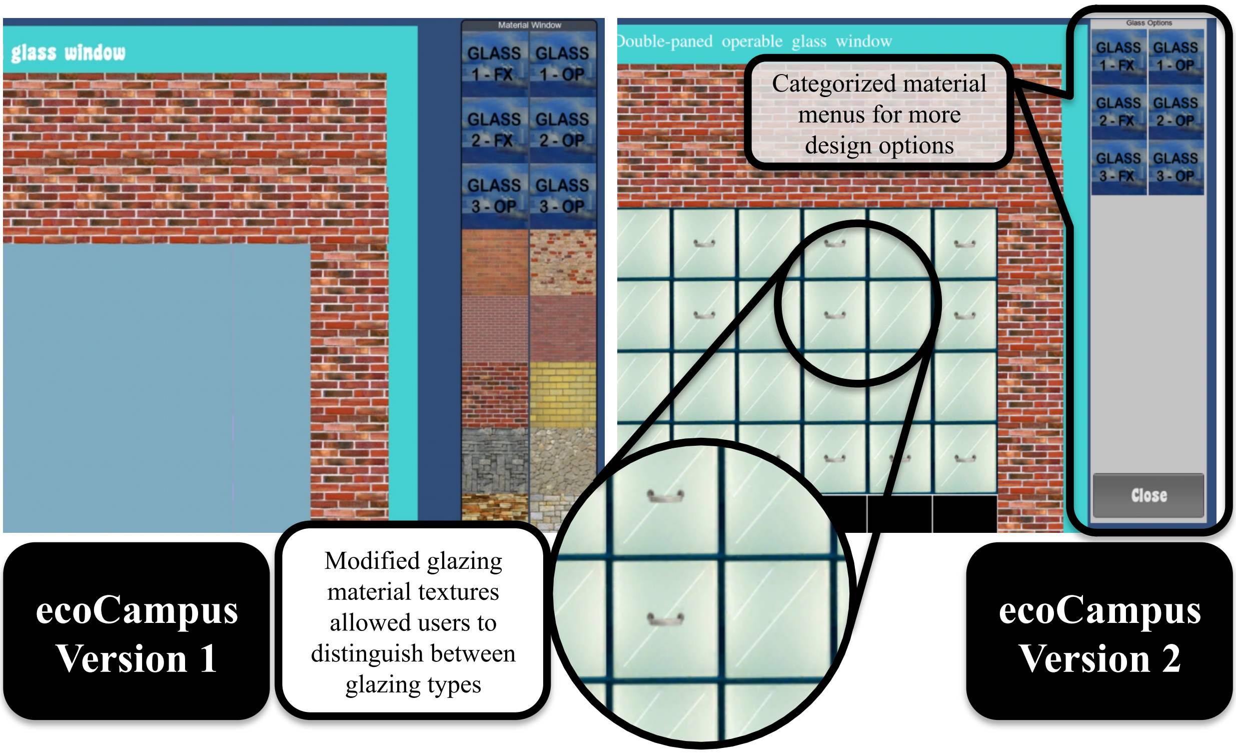
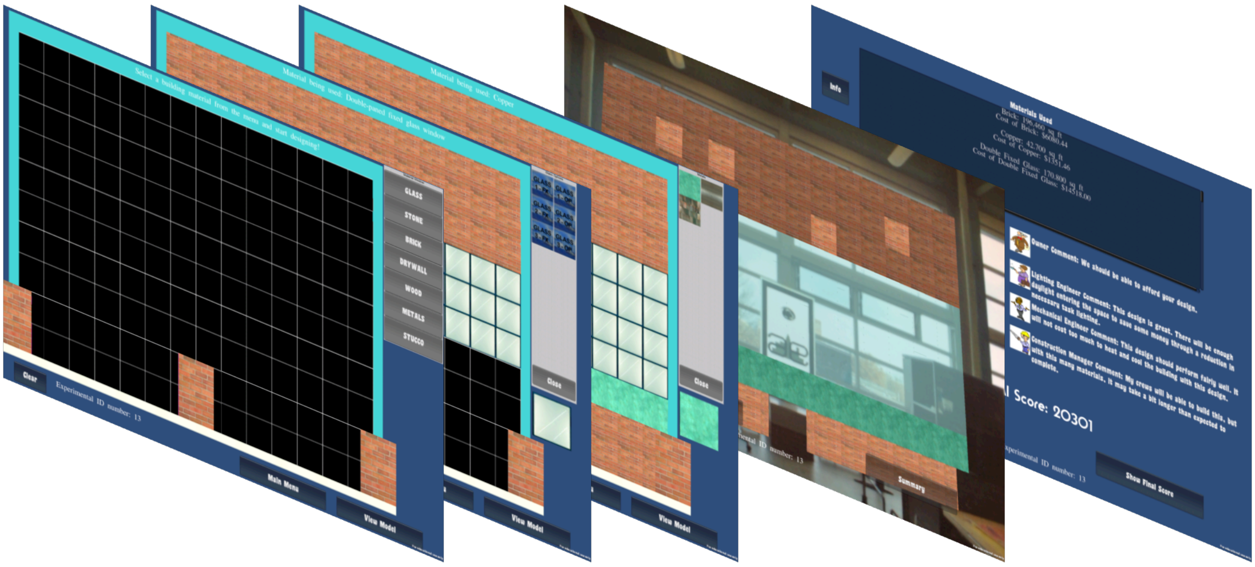


Figure 5: The original design interface (left) was modified for the second version (right) to allow for more material choices, faster design creation, and visual distinction between glazing options.

From the fall 2012 semester, students also voiced concerns with the inability to distinguish between glazing types in the first version of ecoCampus. In the first version, single, double, and triple-pane glazing options were all rendered the same in the design interface. In the second version of ecoCampus the material textures assigned to the different glass options were modified to allow a user to quickly determine how many panes of glass a certain window would have based on the number of diagonal slashes through the window image. Additionally, handle icons were added to the operable glass pieces, which allowed users to quickly determine if a fixed or operable option had been currently selected for a given piece of geometry, which is shown in Figure 5.

Several of the students in the initial implementation also suggested that the application should have more realistic design feedback. In the first version students were provided with three virtual project critics that would generate performance feedback related to initial cost, daylight usage, and lifecycle cost performance in the simulation interface. To address this suggestion, the design summary user interface was expanded to include an additional virtual critic to encourage students to consider constructability. This construction manager critic was added to the simulation interface to encourage students to recognize the value of simple designs for constructability.

The modified ecoCampus workflow can be seen in Figure 6. While certain shortcomings of ecoCampus still exist, the methodology described for developing the tool was successful in creating a functional prototype. Future development of the tool will continue to follow the same development process for enhancing the application. The authors suggest that for application development of augmented reality-based simulation games, the same process can be used for efforts in other educational disciplines as well.



Select a building material from the menu and start designing!

Material being used: Double-paned fixed glass window

Material being used: Copper

Materials Used
Brick: 196.460 sq ft
Cost of Brick: $6080.44
Copper: 42.700 sq ft
Cost of Copper: $1351.46
Double Fixed Glass: 170.800 sq ft
Cost of Double Fixed Glass: $14518.00

Owner Comment: We should be able to afford your design.

Lighting Engineer Comment: This design is great. There will be enough daylight entering the space to save some money through a reduction in necessary task lighting.

Mechanical Engineer Comment: This design should perform fairly well. It will not cost too much to heat and cool the building with this design.

Construction Manager Comment: My crews will be able to build this, but with this many materials, it may take a bit longer than expected to complete.

l Score: 20301

Experimental ID number: 13

Figure 6: The ecoCampus workflow was modified from the first version to incorporate feedback obtained from the first implementation.

4. CONCLUSIONS

This work has discussed the workflow that was used to develop the educational AR-based simulation game called ecoCampus. This tool was used to teach engineering students about basic building sustainability principles related to an exterior wall design. The process used for developing this tool has been presented in this paper and consists of several steps including:

- Identifying educational objectives;
- Developing a list of functional requirements;
- Developing a workflow for the educational design module;
- Determining the development infrastructure for creating the application;
- Developing the tool; and
- Refining the initial prototype through iteration.

The development process presented offers a workflow that can successfully lead to a functional prototype application as evidenced by the successful creation of ecoCampus. It is important to note that the developed ecoCampus prototype does have certain limitations, which were identified during the application testing steps described. While some of these limitations could be addressed in future developments of the tool, the suggested development process was effective at providing an opportunity to clearly identify these deficient aspects of the tool. Even with some minor performance limitations, the ecoCampus prototype created from the suggested development process did work from a technical point of view and it also proved to be beneficial from a pedagogical point of view for the students who used it (Ayer et al., 2013a).

The methodology presented for development is not intended to be interpreted as the single best method for creating this type of application as it was not directly compared to other possible developmental strategies. Instead, the contribution of this work is in illustrating the process and components used for ecoCampus to allow future developers to avoid the pitfalls observed during ecoCampus development and to also allow future work to have a functional development plan from the beginning of a project. This can allow future developers to extend this type of educational application to achieve additional benefits that were not targeted in the initial ecoCampus prototypes and also explore how this type of educational approach may be able to offer benefits to other educational disciplines.

5. ACKNOWLEDGEMENTS

The authors of this work would like to thank: the Raymond A. Bowers Program for Excellence in Design and Construction of the Built Environment for financially supporting this work; Zach Stoltz and Chris Hubert for their contributions toward the development of the ecoCampus prototypes; and all of the students who participated in this research who helped to demonstrate the strengths and weaknesses of this new educational approach.

6. REFERENCES

Aldrich, C. (2004). Six Criteria of an Educational Simulation, *Retrieved on February 2007 from: http://www.learningcircuits.org/NR/rdonlyres/F2ED000A-7A59-4108-A6CB-1BE4F4CC1CA5/4719/clark_e2.pdf.*

Aldrich, C. (2005). *Learning by Doing. A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in e-Learning and Other Educational Experiences*, John Wiley and Sons, Inc., Pfeiffer, San Francisco, CA.

Ayer, S. K., Messner, J. I., and Anumba, C. J. (2013a). ecoCampus: A New Approach to Sustainable Design Education, *Proceedings of the 13th International Conference on Construction Applications of Virtual Reality*, October 30-31, London, U.K.

Ayer, S. K., Messner, J. I., and Anumba, C. J. (2013b). Challenges and Benefits of Open-Ended Sustainable Design in First Year Engineering, *Journal of Professional Issues in Engineering Education and Practice, ASCE*, 140(2).

Ayer, S. K., Messner, J. I., and Anumba, C. J. (2013c). Assessing the Impact of Using Photographic Images to Influence Building Retrofit Design Education, *Proceedings of the Architectural Engineering Institute 2013: Building Solutions for Architectural Engineering*, April 3 - 5, University Park, PA, USA.

Azuma, R. T. (1997). A Survey of Augmented Reality, *Presence: Teleoperators and virtual environments*, 6(4), 355–385.

Bartle, R. (2003). *Designing Virtual Worlds*, New Riders Publishing, United States of America.

Beck, K. (1999). Embracing Change with Extreme Programming, *Computer*, 32(10), 70–77.

Boehm, B. W. (1988). A Spiral Model of Software Development and Enhancement, *Computer*, 21(5), 61–72.

Brown, J. S., Collins, A., and Duguid, P. (1989). Situated Cognition and the Culture of Learning, *Educational Researcher*, 18(1), 32–42.

Finke, R. A. (1990). *Creative Imagery: Discoveries and Inventions in Visualization*, Hillsdale, NJ, Lawrence Erlbaum Associates.

Finke, R. A., Ward, T. B., and Smith, S. M. (1996). *Creative Cognition: Theory, Research, and Applications*, Cambridge, MA.: MIT Press.

Furmanski, C., Azuma, R. T., and Daily, M. (2002). Augmented-Reality Visualizations Guided by Cognition: Perceptual Heuristics for Combining Visible and Obscured Information, *International Symposium on Mixed and Augmented Reality*, IEEE, Malibu, CA, 215 – 320.

Gottesdiener, E. (1995). RAD Realities: Beyond the Hype to How RAD Really Works, *Application Development Trends*, 28 – 38.

Gredler, M. (1994). *Designing and Evaluating Games and Simulations : A Process Approach*, Gulf Pub. Co., Houston.

Jacobs, J. W., and Dempsey, J. V. (1993). Simulation and Gaming: Fidelity, Feedback, and Motivation, *in Interactive Instruction and Feedback*, J. V. Dempsey and G. C. Sales, eds., Englewood Cliffs, NJ: Educational Technology Publications, 197–227.

Johnson, S. (1997). What's in a Representation, Why Do We Care, and What Does It Mean?, *Representation and Design*, Examining the Evidence from Psychology Proceedings of ACADIA '97.

Larman, C. (2003). Iterative and Incremental Development: A Brief History, *Computer*, 36(6), 47–56.

Lave, J., and Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Press Syndicate of the University of Cambridge, Cambridge, UK.

Luqi. (1989). Software Evolution Through Rapid Prototyping, *Computer*, 22(5), 13– 25.

Milgram, P., and Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays, *Transactions on Information Systems*, E77-D(12), 1321 – 1329.

Rieber, L. P. (1994). Visualization as an Aid to Problem-Solving: Examples from History, *Proceedings of the National Convention of the Association for Educational Communications and Technology*, Nashville, TN.

Royce, W. W. (1970). Managing the Development of Large Software Systems, *IEEE WESCON.*