Simulation Design and Development for Learning Seasons and Lunar Phases using HTML5 and JavaScript

Michelle Zhu, Sneha Gulati and Nicole Panorkou
Montclair State University
zhumi, gulatis1, panorkoun@montclair.edu

Abstract - **This paper presents our recent work in progress aiming to design effective learning simulations of day/night and seasons and lunar phases using web technology for K-12 Earth and Environmental science curriculum. Two interactive simulations using HTML5, JavaScript and CSS are developed with steerable parameters for students to interact and manipulate. The design and development details are discussed in this paper. We are currently in the process of designing the lesson plans with various investigations to engage students with the seasons and lunar phases concepts as well as assessments to evaluate their learning outcomes.**

Index Terms – Computer Simulations, Earth and Environmental Science, HTML5, JavaScript

Introduction

Earth and Environmental Science (EES) studies the earth's dynamic history. It offers a rich context for investigating STEM education as it is a multidisciplinary field integrating almost every discipline of science. Our goal in this study is to develop students' understanding of EES on some important and complex earth science concepts in the solar system which pose a great challenge for teachers to convey effective teaching in the traditional classroom using static representations, such as text description and illustrative figures.

To engage students in dynamic representations of EES concepts, we already developed several simulations, such as the water cycle and the solar system, using techniques such as parameter steering, attractive animation and rich input and output plots [1]. Recently, we are designing two new simulations using web technologies to avoid software installation on students' computers since many schools only have chromebooks that do not allow the installation of software. This paper describes these two latest examples of our simulation design, namely day/night and seasons and lunar phases that would be integrated into two EES modules one studying day/night and seasons, and one exploring the lunar phases.

Our curriculum modules' design follows the Next Generation Science Standards, the Common Core State Standards for Mathematics and the New Jersey student learning standards. For the design of the specific simulations and modules, we focus specifically on the following NGSS standards [4], in which students:

"ESS1-1 Use observations of the sun, moon, and stars to describe patterns that can be predicted. Clarification Statement: Examples of patterns could include that the sun and moon appear to rise in one part of the sky, move across the sky, and set; and stars other than our sun are visible at night but not during the day."

"1-ESS1-2 Make observations at different times of year to relate the amount of daylight to the time of year. Clarification Statement: Emphasis is on relative comparisons of the amount of daylight in the winter to the amount in the spring or fall."

"**MS-ESS1-1** Develop and use a model of the Earth-sun-moon system to describe the cyclic patterns of lunar phases, eclipses of the sun and moon, and seasons. Clarification Statement: Examples of models can be physical, graphical, or conceptual. "

For our design, we adopt *"project-based learning"* (PBL), a pedagogy that organizes learning around projects which are often complex tasks which require students to perform problem-solving, decision making, investigation activities over extended period of time culminating a product or presentations [2,3]. The design is currently undergoing incremental improvements and the final products will be released for community use soon.

Theoretical Framework and Technologies

For our design, we incorporate the 5E or called "Five Es" instructional model, which is developed by the Biological Science Curriculum Study (BSCS), a team led by Principal Investigator Roger Bybee in the future lesson plan [5]. The 5E represents the five stages in the teaching and learning, namely, Engage, Explore, Explain, Extend (or Elaborate), and Evaluate. The interactive simulation design particularly emphasizes the three Es: Engage, Explore and Elaborate.

Engage: The simulations will be interactive and responsive in order to capture student's interests and get them involved in the simulation play.

Explore: The simulations will update based on the chosen input parameters and students can construct their own understanding and modify existing misconceptions on the concepts. Guided simulation interactions will be designed to help students explore the simulations from various perspectives.

Elaborate: Based on the new knowledge constructed, students would be able to explain and make connections with other related problems. We will include specific prompts in our investigations to help student make these connections.

The following programming technologies are used:

HTML: Hypertext Markup Language, a standardized system for tagging text files to define font, color, graphic, and hyperlink effects on World Wide Web.

JavaScript: It is a client-side scripting language executed with html on user browser or client device. The purpose of javascript is to make the content of html dynamic.

jQuery, jQuery-UI: These are the lightweight Javascript libraries. Purpose of jQuery library is to make Javascript to be used conveniently on browser. jQuery provides a wide range of methods which make code less complex.

CSS and CSS Animations: It is a language for describing the presentation of web pages.

Open Source REST API for Real-time Weather: This Application Programming Interface (API) provides the real-time weather data for different cities around the world. We have a button for user to click in order to see the current weather of the city he/she selects. The computer needs to have Internet access in order to enable this function. https://openweathermap.org/api

Webservers: Primary purpose of web server is to store, process and deliver web pages to clients. The communication between client and server exploits HTTP protocol.

SIMULATIONS DESIGN

I. Day/Night and Seasons

The purpose of this simulation is for students to explore the patterns of the earth and sun rotation and investigate how these patterns cause day/night as well as seasons due to the amount of sunlight received. The day/night and seasons simulation is a web-based application which consists of an animation of the earth revolving around the sun with interactive controls. Application is completely designed by HTML5, Javascript (jQuery) and css (css animation), can be deployed on any webserver and accessed via web-browser.

Fig. 1 demonstrates the architecture of this simulation. There is a request from browser to webserver to provide the animation content with the current selected parameters. Once the request arrives to the webserver, the request is processed and the html page returns to the client browser. If any API call is issued, such as real-time weather services, the webserver passes the call to the remote REST web service for data. Once the data returns from the web service, HTML dynamically populates the data on the client webpage using Javascript API. On the contrary,the predicted weather is static information stored in a table and can be easily retrieved without any third-party webservice call.

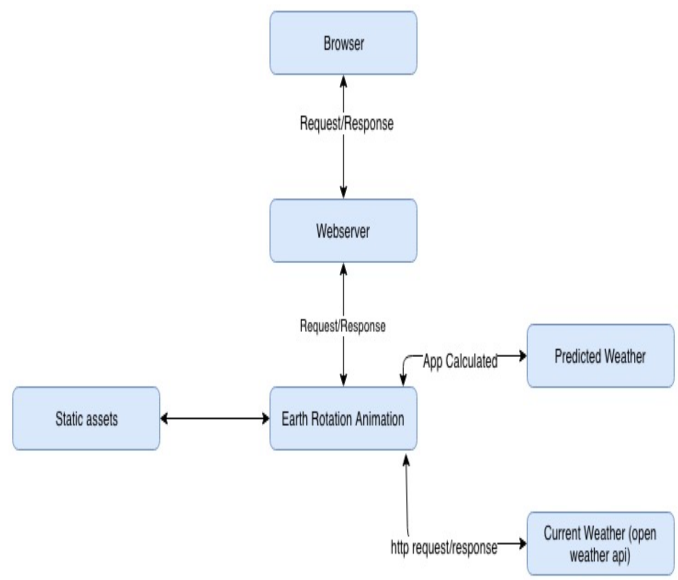

FIGURE 1
THE ARCHITECTURE OF OUR SEASONS SIMULATION

The Day/Night and Seasons simulation predicts the typical temperature based on the selected city in a selected month based on historical data. The open weather API fetches the current weather from the remote webservices. Our system currently can handle some geographically representative cities including San Francisco, New York, Anchorage, Sydney and Nairobi. The jQuery-based thermometer is used to show the temperature with JSON specifying the value of the temperature. The core html structure is defined in index.html which includes multiple other files including css files, jQuery, thermometer.js, jquery-ui.js and index.js.

Runtime css manipulation is done at the Javascript level, while the first load of the application browser renders the components via css. Once the application is loaded, Javascript manipulates the css and re-arranges the components and its state on the browser.

As shown on the interface of Fig. 2, the Control panel of our system includes the following:

- Speed slider to control the speed of earth rotation.
- Drop-down list to select the earth tilting angle between 0 and 23.5 degrees. If the user chooses 0

tilting degree, it will show that there will be no seasons on the earth. The weather will be like spring or autumn throughout the year.

- Drop-down list to change the desired city and month of the year.
- Button to show the real-time weather for the selected city at current time.
- Button to show the predicted weather for the selected city and selected month.

By exploring our simulation, students will be prompted to explain the role of the rotation of the Earth on its axis for causing the phenomena of day and night and seasons.

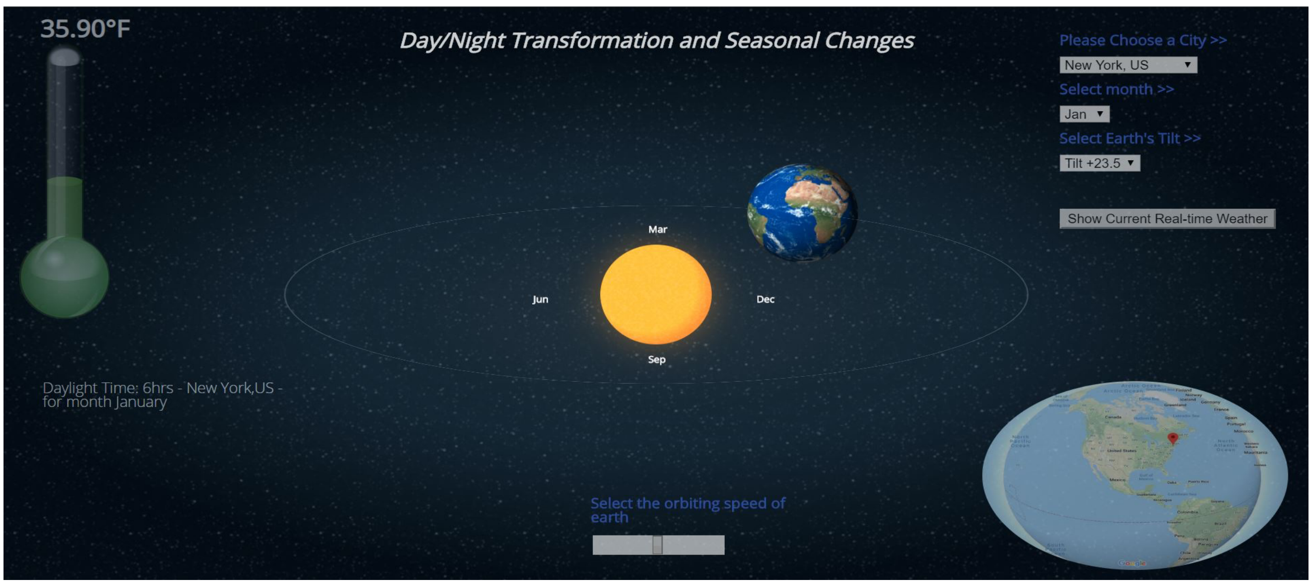

FIGURE 2
THE INTERFACE FOR DAY/NIGHT AND SEASONS

Students will also explore how the tilt of the Earth on its axis influences a) the length of the day and night at different hemispheres on Earth and b) the average seasonal temperature at the different hemispheres on Earth. Students will use the simulation to form relationships about the length of day and night at different seasons and at different hemispheres on Earth as well as relationships between the average seasonal temperature at the various hemispheres on Earth.

II. Lunar Phases

The purpose of this simulation is for students to describe how the Earth and the moon are positioned in relation to the sun, explain why there are different phases of the moon as well as lunar and solar eclipses.

Through the exploration of the simulation, students can not only visually explore the solar system but also connect to their daily observances. Thus, they can engage in developing hypotheses, formalizing ideas, and analyzing the results. In terms of math, students will also understand the concept of variables and their interrelationships.

In the system architecture of the simulation in Fig. 3, the browser sends a request to the webserver to draw the animation page. Once the webserver gets the request, it starts to process it. Compared with day/night and season simulation, the lunar phase one does not include any real-time webservice call and is simpler in the execution path. The response after look-up will be returned to client browser.

As shown in the screenshot of the interface Fig. 4, the user can choose different phases of the moon including New Moon, Waxing Crescent Moon, First Quarter Moon,

Waxing Gibbous Moon, Full Moon, Waning Gibbous Moon, Third Quarter Moon, and Waning Crescent Moon. The description and the picture of each type of moon phase will be shown and the orbiting location of the moon will also be correctly displayed in the interface. We will add controls to illustrate the mechanisms of solar and lunar eclipses soon.

CONCLUSIONS

This paper describes our work in progress of designing two simulations for the solar system. Currently, we are in the process of designing the lesson plans. After the modules are designed, they will go over an iterative process of refinement with student and teacher's feedback. We will test the modules with students, monitor students' reasoning and modify the lessons and simulation interface accordingly. Our simulation design is very important to inspire active student engagement and initiate exploration and discussion around various EES topics taught in K12 sciences. We also aim to instill mathematical and computational thinking by including investigation and assessment questions on related math and programming assignments.

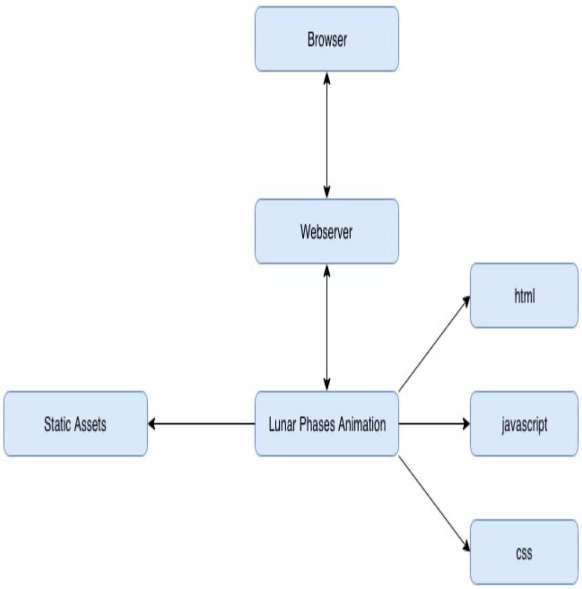

FIGURE 3
THE ARCHITECTURE OF OUR LUNAR PHASES SIMULATION

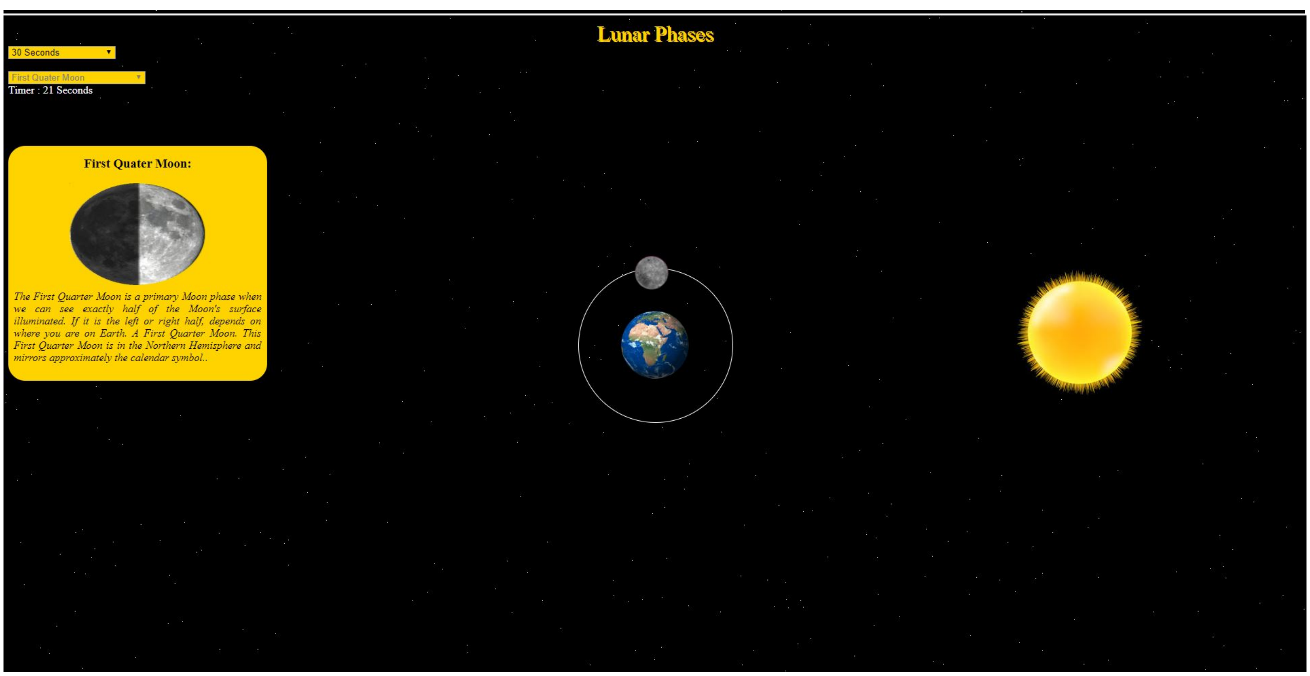

FIGURE 4
THE SCREENSHOT OF THE INTERFACE FOR LUNAR PHASES

ACKNOWLEDGMENT

This research is supported through a STEM+Computing grant from the Division of Research on Learning of the National Science Foundation (# 1742125).

REFERENCES

[1] Zhu, M., Panorkou, N.,, Lal, P., Etikyala, E., Germia, P. Iranah, B. Samanthula B., and Basu, D. Integrating Interactive Computer Simulations into K-12 Earth and Environmental Science. *Proceedings of IEEE Integrated STEM education Conference (ISEC) 2018*, pp. 220-223, Princeton, NJ. March 10th, 2018. doi: 10.1109/ISECon.2018.8340488

[2] Jones, B. F., Rasmussen, C. M., & Moffitt, M. C. (1997). Real-life problem solving.: A collaborative approach to interdisciplinary learning. Washington, DC: *American Psychological Association*

[3] Thomas, J. W., Mergendoller, J. R., and Michaelson, A. (1999). Project-based learning: *A handbook for middle and high school teachers*. Novato, CA: The Buck Institute for Education

[4] National Science Teacher's Association with Next Generation Science Standards: https://ngss.nsta.org/

[5] Bybee, R.W., J.A. Taylor, A. Gardner, P. Van Scotter, J. Carlson Powell, A. Westbrook, and N. Landes. 2006. BSCS 5E instructional model: Origins and effectiveness. *A report prepared for the Office of Science Education, National Institutes of Health*. Colorado Springs, CO: BSCS.

AUTHOR INFORMATION

Michelle Zhu, Associate Professor, Department of Computer Science, Montclair State University.

Sneha Gulati, Research Assistant, Department of Computer Science, Montclair State University

Nicole Panorkou, Associate Professor, Department of Mathematical Sciences, Montclair State University.