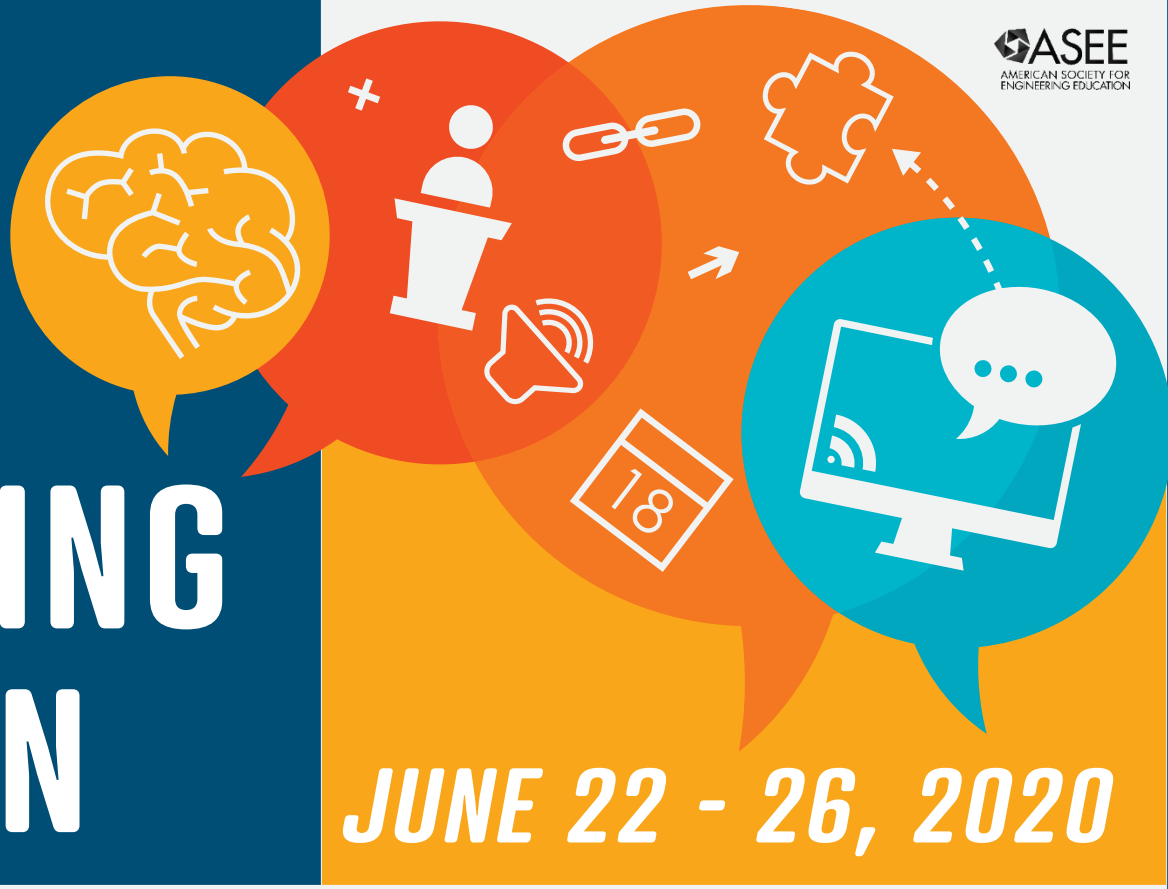
AT HOME WITH ENGINEERING EDUCATION



Asee's Virtual Conference

#ASEEVC

A simple and efficient markup tool to generate drawing-based online assessments

Nicolas Nytko[1]

Matthew West[2]

Mariana Silva[1]

[1] Department of Computer Science, University of Illinois at Urbana - Champaign

[2] Department of Mechanical Science and Engineering, University of Illinois at Urbana - Champaign

Background

- Online assessments using PrairieLearn
  - Learning Management System developed at University of Illinois
- Proctoring with Computer Based Testing Facility (CBTF)
  - Students may self-schedule exams, quizzes
  - In past three years:
    - "50,000 exams for over 6,000 unique students in 25–30 classes"[1]

[1] C. Zilles, M. West, G. Herman, and T. Bretl, Every university should have a computer-based testing facility, in Proceedings of the 11th International Conference on Computer Supported Education (CSEDU 2019), 2019.

PrairieLearn Background

- Questions are defined as "problem generators"
- Problem generators create questions with randomized values and parameters
- Defined by:
  - HTML file for display
  - Python server code for randomization

HW1.1. What is 2+2?

What is $2 + 2$?

$2 + 2 =$ number (rtol=0.01, atol=1e-08)

Save & Grade Save only

Homework 1

Assessment overview

Total points: 0/1

Score: 0%

Question

Value: 1

History:

Awarded points: 0/1

Report an error in this question

Previous question

What is $2 + 2$?

$2 + 2 =$ number (rtol=0.01, atol=1e-08)

```
1  <p> What is $2 + 2$? </p>
2  <pl-number-input correct-answer="4"  answers-name="answer"
3                   label="$2+2=$" ></pl-number-input>
4
```

Question Randomization

- How can we put random values and logic into HTML?
  - Answer: Generate in Python and use templating
- Mustache templating allows setting values and conditional logic
- Entire sections of markup can be enabled or disabled at render time

What is $7 + 5$?

$7 + 5 =$ number (rtol=0.01, atol=1e-08)

server.py

```python
import numpy as np


def generate(data):
    x = np.random.randint(1, 10)
    y = np.random.randint(1, 10)
    z = x + y

    data['params']['x'] = x
    data['params']['y'] = y
    data['correct_answers']['z'] = z

```

question.html

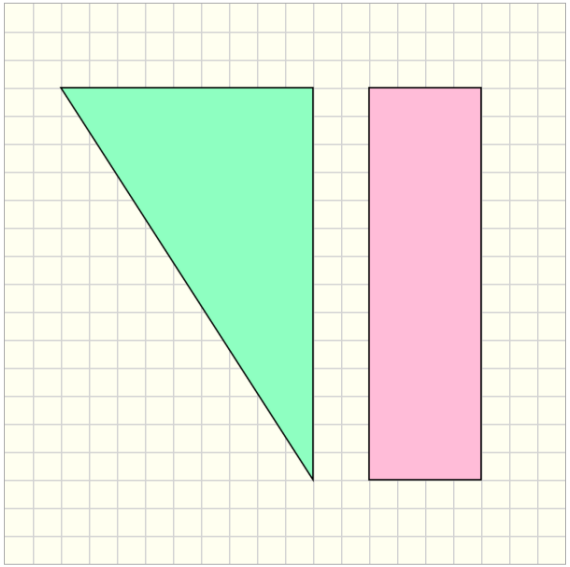
```
<p> What is ${{params.x}} + {{params.y}}$? </p>
<pl-number-input answers-name="z"
  label="${{params.x}}+{{params.y}}=$">
</pl-number-input>

```

Drawing Questions

- Above system works well for regular multiple choice and short answer questions
- How can we extend this to more open drawing and sketching questions?
  - Sketching Free-Body Diagrams
- Answer: we specify drawing questions as markup, too!

"Mark the center"



Graded Objects

Help Buttons

(The expected tolerance is 1/2 square grid for position and 10 degrees for angle.)

```
<pl-drawing grid-size="20" answers-name="centroid"
  width="400" height="400" gradable="true">
  <!-- Answer -->
  <pl-drawing-answer>
    <pl-point x1="300" y1="200"></pl-point>
    <pl-point x1="160" y1="153"></pl-point>
  </pl-drawing-answer>

  <!-- Initial drawing state -->
  <pl-drawing-initial>
    <pl-triangle x1="40" y1="60" x2="220" y2="340"
      x3="220" y3="60" color="green1"></pl-triangle>
    <pl-rectangle x1="300" y1="200" width="80" height="280"
      color="pink1"></pl-rectangle>
  </pl-drawing-initial>

  <!-- Controls -->
  <pl-controls>
    <pl-controls-group label="Graded Objects">
      <pl-drawing-button type="pl-point"></pl-drawing-button>
    </pl-controls-group>
    <pl-controls-group label="Help Buttons">
      <pl-drawing-button type="delete"></pl-drawing-button>
      <pl-drawing-button type="help-line"></pl-drawing-button>
    </pl-controls-group>
  </pl-controls>
</pl-drawing>

```

Drawing Question Randomization

- Randomize drawing-based questions in same spirit as "normal" questions
- Generate random coordinates, angles, etc. in Python and template on HTML
- Can use grouping functionality to enable or disable whole groups of elements
    - Provide different "variants", for example

Cantilever Beam FBD

Shear and Bending Moment Diagrams

No External Software

- Software is web-based using standard HTML canvas
  - Can run on anything with a web browser: PC, Mac, Linux, even mobile
- To define their answer, users may drag and drop elements onto canvas
- Submitted as normal question in PrairieLearn and students can get immediate feedback