1. Project Scope

1.1 Project Title

Qwizm, a client-side assessment system with randomized numerical inputs and immediate marking of multiple-answer questions accepting numerical entries.

1.2 Capstone Category and Related courses

Category: Web-development

Related courses:

- DGMD E-20  Developing Interactive Media
- CSCI E-3  Web Programming/JavaScript
- DGMD E-27  Modular and Mobile Front-End Design II
- DGMD E-60  Applied Online Course Design

1.3. Project Goal

1.3.1 Project Overview

The goal of this capstone is to produce efficient and concise code for a website that contains quizzes, or assessments. Quizzes are comprised of an arbitrary number of questions. Each question has individualized\(^1\) input values and is comprised of an arbitrary number of required numerical answers. These questions, each requiring multiple answers, are suitable for step-by-step progression through a typically complex take-home or in-class assessment problem set commonly found in introductory engineering technology courses such as Structural Statics, Strength of Materials, or Fluid Mechanics.

Assessments comprised of questions having randomized inputs and requiring a numerical answer are a common feature of many learning management systems (LMSs). These question-types are referred to by a variety of names: ‘Calculated’ question type in the Moodle LMS (Moodle Question Types, 2018), ‘Formula’ question type in the Canvas LMS (Team, 2019) or ‘Calculated Formula’ question type in the Blackboard LMS (Calculated Formula Questions, 2018). Unfortunately, each of these LMS questions marks only a single numerical answer.

\(^1\) Generally, every student gets a different set of input values to each question. As there are not infinitely many combinations of input values, repetition is possible—but rare.
Frequently, though, questions require more than a single answer; a simple example of this is calculating the three angles of a triangle. Or longer, more complex, questions—which are typical in engineering courses—may need a series of steps that progress towards a final answer; in these cases, it is helpful—or even essential—for students to get feedback as they proceed through a lengthy set of calculations.

Consider a typical fluids question shown below:

![Diagram of fluid system](image)

**Q4:** All elbows are standard. Disregard entrance and exit losses, and losses at junctions B and C. Pipes 1 and 2 each contain a fully open globe valve and pipe 3 contains an open angle valve. Determine the flow Q through the system in L/s.

This question requires 40-50 lines of calculation to find the requested value, Q. To expect a student to arrive at the numerically precise correct answer without intermediate feedback is not always realistic; by breaking the question down into multiple parts, and by giving immediate feedback on these parts—marking entries correct or incorrect—students are encouraged and guided towards successful completion of the question.

Moodle (Moodle, 2019) is a popular open-source LMS used by many major educational institutions; it has around 170 million users worldwide (Moodle Statistics, 2019). It offers an impressive list of question types (Moodle Question Types, 2018) but no randomized-input multiple-answer question type.

There is a plug-in (Formulas Question Type, 2014), maintained separately from the Moodle core, that does provide the required multiple-answer question functionality. If your institution uses Moodle and your LMS administrator will install the necessary plug-in, then this ‘Formulas’ question type is currently your best option. (Note that the number of sites that use this plug-in is a relatively small 547.)

Despite Moodle having 170 million users, this only represents 12% of student enrolments worldwide (Fenton, 2018). Few students have access to this Formulas question type. Most students do, however, have access to a web-browser. My proposed Qwizm project aims to provide this multiple-answer question functionality independent of LMS, on a website freely accessible to a standards-compliant browser.
In no way does this project claim to replace an LMS; it will not store student records on a server and, as a result, cannot offer automatic mark entry into a gradebook. It does offer some functionality that even the Moodle ‘formulas’ plug-in mentioned above does not: more convenient checking of entered answers; and dynamic overlaying of randomized input values on top of a static problem drawing or graphic.

**Why Qwizm rather than host the project functionality within an LMS?**

It is not simple, or even possible in many cases, to insert a functioning .html site containing JavaScript into an LMS. The LMS I have most experience with, Desire2Learn Brightspace (D2L, 2019), automatically strips out all JavaScript to the extent that one cannot even include simple JavaScript libraries in pages. There is a Moodle plug-in (Richter, 2019) for inserting ‘JavaScript activities’ into Moodle and I have used it for the simple viewing and hiding of windows (Morgan, Guided Practice, 2018); even in this simple use case, the interface for entering code into Moodle—without the intellisense, syntax highlighting and live browser updating available in modern text-editors—complicates content creation. I would not want to write a full application under those conditions. This plug-in is maintained by a single volunteer. Furthermore, updates to LMSs can break existing plug-ins. Plug-ins are written by individuals or small teams and there is no guaranteed forward compatibility—people change careers, have a family, retire, lose interest… For example, the maintainer of Moodle’s Formulas plug-in has now retired and is no longer able to perform the maintenance and the current status of the project is “seeking for adoption,” according to messages at the bottom of the Formulas question type page (Formulas Question Type, 2014).

I would like the project to be as accessible to as many users as possible. One standards-compliant site seems the easiest way for me to do that. In terms of personal motivation, new tools make web programming more fun and I am likely to produce more enthusiastically when not frustrated by the constraints of coding in a single dialog window within an LMS.

**1.3.2 Prior Work**

It should be emphasized here that the goal is to produce refactored code for the framework of my existing website for Qwizm (Qwizm, 2012), developed largely as a proof of concept about a decade ago. The production of a substantial amount of content (i.e., question sets) is not an objective of this capstone—although the content for a quiz containing more general math and trigonometry questions will be created so that those with no specific engineering knowledge will be able to experience, and test, the functionality of Qwizm. Note that it is not possible for users to modify questions or quizzes hosted on Qwizm. (This limitation is to be addressed by a future project described in section 1.3.4 below.)
1.3.3 Capstone Development
Existing utility functions will be refactored. For example, the generation of randomized question input variables using a custom-built pseudo-random number generator; the conversion of numerical values of a specified number of significant digits for question statement display and for marking correct or incorrect; automatic update of a summary table showing student answers and marks awarded so far.

The new Qwizm site will be responsive, taking advantage of developments in CSS over the last decade. A particular challenge may be maintaining the location of individualized question inputs in precise position, on top of the static question image, for different devices and browser window sizes. LocalStorage in the browser will be used to maintain the state of each quiz between browser sessions. A utility to handle encryption of numerical variable contents, using a simple cypher based on student ID and question ID so that correct answers cannot be simply extracted from the code using a browser’s developer tools, will be developed.

Existing and new code will be written with the development of an authoring tool (described in 1.3.4 below) in mind.

1.3.4 Going Forward
This proposed Qwizm capstone is the first of two distinct—although related—projects. The code for this first project, Qwizm, will be freely accessible from Github (Morgan, nuQwizm, 2019) for those who want to download the code and extend the question sets and topics, create their own quizzes, and then host the results themselves. This will require familiarity with web-site maintenances and some knowledge of JavaScript—skills not always possessed by instructors and teachers.

The second project will be a desktop application, to be named Qwizard, that aims to make the creation of quizzes and questions more accessible. Qwizard will be a form-based editing tool for the creation and modification of quizzes and questions that, although requiring the ability to build a formula of the type frequently used in a spreadsheet, does not require specific programming knowledge. Qwizard will be a desktop application, using HTML5 syntax in Electron (Electronjs, 2019). A desktop application will more easily enable users to create, store and edit a personal library of questions and quizzes on their local computer or in cloud storage. Qwizard will use the HTML, CSS and JavaScript code developed for Qwizm to generate a quiz in a single self-contained .html file containing all the resources required to complete a quiz. This file can be hosted on the instructor's own site, on an institution shared drive or distributed directly to students who can open the quiz in a standards-compliant browser.

Note that Qwizard is beyond the scope of this Qwizm capstone project.
1.4. Learning Goals
I have a rudimentary knowledge of most of the skills that I require for this project, gained from previous courses taken at the Extension School. There is a list of skills that I need to refresh and extend, of course: using LocalStorage; mobile-first responsive design (although the nature of complex questions prevalent in Qwizm requires pencil, paper and calculator work, mobile is still sufficient for answer input); SASS and CSS3; modern JavaScript techniques and a wealth of other knowledge that I haven’t considered yet. There will be something new to learn every day, I am sure.

1.5. Elevator Pitch
Qwizm: A readily accessible tool that can be used by instructors and teachers for assessments comprised of complex multiple-answer questions—and that also reduces the marking load; a tool that can also be used by students for self-assessment and review to gain mastery of content prior to a test or exam. Using modern web technologies, Qwizm makes all this is available through your browser, a single click away.

1.6. Target Audience
The number of potential users for this capstone project, essentially a single website http://qwizm.org, will be dependent upon the amount of useful content (questions and quizzes) available upon the site.

My long-term goal is to build out the content. Existing questions in the initial version of Qwizm will be rewritten. A complete set of questions and quizzes is planned for a course in Structural Statics, to be followed by building fresh content to add to my current Strength of Materials and Fluid Dynamics/Water Resources questions.

The new version of Qwizm that results from this capstone will make question generation more streamlined and efficient, enabling the volume of content to grow more quickly. Completion of the previously mentioned Qwizard application will further increase efficiency and ‘reach’ of the project.

With the amount of content so far created, student users are limited to those who are taking the Strength of Materials and Water Resources courses at my former institution—currently less than 200 per year. As more content is brought online and ‘marketed’ on Merlot.org (Merlot, 2019), and as more instructors recognize the potential, this number is expected to grow.

There were more than 100,000 graduates with engineering degrees in the US in 2014-2015 (Yoder, 2019) and each of these students could potentially have made use of the Qwizm site at some point during their studies. Engineering technologists with two-year diplomas, my primary target, will also benefit. (Yoder, 2019)
<table>
<thead>
<tr>
<th>Country</th>
<th>Engineering Graduate Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>106,658</td>
</tr>
<tr>
<td>Canada</td>
<td>6106</td>
</tr>
</tbody>
</table>

*Figure 1 Annual engineering graduates in the US and Canada*

Notes:

1. Most but not all engineering disciplines include the introductory courses (Statics, Strength of Materials, Fluid Mechanics/Water Resources) where I intend to focus my content creation so the number of potential users is less than described above.
2. Engineering technologist students, who would benefit from access to Qwizm, are not included in these figures, would increase the potential number of users.
3. Very few (<200) are currently aware of the product. With the creation of more content and marketing Qwizm on Merlot (Merlot, 2019) will, hopefully, increase these figures dramatically.

I am fortunate that I have a reasonable idea of how users are going to respond when completing an assignment on Qwizm through the fielding of queries and/or complaints relating to quizzes that I have assigned to several cohorts of students. I have been able to incorporate usability fixes into the existing site as issues arose. Some issues I have chosen not to ‘fix’: a common complaint is that Qwizm requires too accurate an answer (entered answers must be accurate to three significant digits—or to four significant digits if the first non-zero digit is a one—and interim calculations to at least five significant digits to reduce the accumulation of numerical errors) but, since this follows the pattern followed for all examples provided in class, and this requirement is clearly and repeatedly stated, I feel it is important that students be able to follow consistently laid out instructions.
**Example Persona: Mature Student (Strong)**

<table>
<thead>
<tr>
<th>Name</th>
<th>John Christian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Douglas dropped out of engineering school at university, worked for in an unrelated field for several decades; he is embracing a late career change, returning to college to study to be a civil engineering technologist. He has good time management skills, is a motivated student, works part time and has two adult children.</td>
<td></td>
</tr>
</tbody>
</table>

| Goals | Since unlimited attempts are allowed at each answer, there is little doubt that Douglas will achieve full marks on this assignment but, due to his other commitments, he would like to get through the quiz as quickly as possible. |
| Frustrations | Minor. Mild annoyance when forgetting to use four significant digits in the answer when the leading non-zero digit is a one. Frustrating by his rusty high-school math expertise. |
| Quote | “I like the immediate feedback available after each answer is submitted. Also, the multiple parts guide you towards the solution, breaking it down into smaller, manageable parts.” |

**Empathy Map:**

<table>
<thead>
<tr>
<th>SAY</th>
<th>THINK</th>
</tr>
</thead>
<tbody>
<tr>
<td>“I’m off to the library to get an early start on this assignment. We have two mid-terms next week, so I want to get this Qwizm wrapped up and out of the way.”</td>
<td>There is a question that looks quite different from the examples done in class. I’d like to figure it out though. Maybe I’ll do the other questions first?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DO</th>
<th>FEEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work on questions until I like I need a break, then spend a bit of time with the kids. That difficult question may become clearer after I’ve slept on it. Hopefully…</td>
<td>I like these quizzes; you know exactly where you are.</td>
</tr>
</tbody>
</table>
**Example Persona: Below Average Student**

<table>
<thead>
<tr>
<th>Name</th>
<th>Lindsay Browne</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goals</strong></td>
<td>Unknown.</td>
</tr>
<tr>
<td><strong>Frustrations</strong></td>
<td>Does not like Qwizm. There are no solutions to the questions online; would much prefer assignments taken directly from problems in the course textbook. Feels that fussing with significant digits is unimportant and a waste of time.</td>
</tr>
<tr>
<td><strong>Quote</strong></td>
<td>“Whatever.”</td>
</tr>
</tbody>
</table>

Empathy map:

<table>
<thead>
<tr>
<th>SAY</th>
<th>THINK</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Has anyone found the answers to question 1 yet? Or built it in a spreadsheet?”</td>
<td>Seven questions and some have more than six parts! I don’t have time for this ****!</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DO</th>
<th>FEEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Will definitely start tomorrow. I know these can take a long time.</td>
<td>This is so much harder than group work.</td>
</tr>
</tbody>
</table>
**Example Persona: Course Instructor**

<table>
<thead>
<tr>
<th>Name</th>
<th>Adam Smith</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course instructor</strong>, having assigned a Qwizm quiz for homework at the end of a course module, wants to demonstrate successful progress through the quiz to increase student confidence.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Goals</th>
<th>To demonstrate to the class that it is possible to get answers marked correct at the first attempt if attention is paid to method and to precision in calculations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frustrations</td>
<td>Student resistance to Qwizm when many would prefer questions from the course text so that everyone has exactly the same question. What should be independent work may become a collaborative effort, resulting in less learning for the weaker students.</td>
</tr>
</tbody>
</table>

| Quote                        | “One of my greatest challenges is to encourage students to challenge themselves to work independently so that they have something of value to contribute to a group when group work is called for.” |

<table>
<thead>
<tr>
<th>Empathy map:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SAY</strong></td>
<td>“Pick a question from the homework set for me to demonstrate the use of Qwizm.”</td>
</tr>
<tr>
<td><strong>THINK</strong></td>
<td>As usual, the stronger students are leading the class and the weaker students are following along. The weaker students will, no doubt, get help from their stronger peers in getting full marks on the homework. But they still need to do their own calculations with their own randomized inputs to each question and write their own solution for submission. What more can I do?</td>
</tr>
<tr>
<td><strong>DO</strong></td>
<td>Work through the chosen question, one part at a time, asking the class for a suggestion about what number to insert in the answer box. If the wrong number is calculated or the wrong number of significant digits are provided by a student, enter the provided number and notice that it is marked incorrect. Ask the class why. Then change the number and show the correct 'check' mark.</td>
</tr>
<tr>
<td><strong>FEEL</strong></td>
<td>Pleased that the class feels better about Qwizm, that the work is not harder than the examples performed in class.</td>
</tr>
</tbody>
</table>
1.7. Metrics, Rubric Table & User Survey

1.7.1 Rubric

Table 1: Success Rubric

<table>
<thead>
<tr>
<th></th>
<th>Minimum Viable</th>
<th>Expected</th>
<th>Good</th>
<th>Excellent</th>
<th>Exceptional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsive Design</td>
<td>Very difficult to use (e.g., navigation and user input) on a small tablet</td>
<td>Possible to use on all device sizes except phone</td>
<td>Possible to use for all device sizes</td>
<td>Excellent to use on all device sizes</td>
<td>Uniquely elegant and responsive</td>
</tr>
<tr>
<td>System Usability Scale (SUS) Score (Brooke, 2013)</td>
<td>50-69</td>
<td>68²-72</td>
<td>73-79</td>
<td>80-89</td>
<td>90-100</td>
</tr>
<tr>
<td>HTML and CSS validation</td>
<td>No major errors</td>
<td>Minimal errors and few warnings</td>
<td>Completely error free and several warnings</td>
<td>Completely error free and few warnings</td>
<td>Completely error and warning free</td>
</tr>
<tr>
<td>Adoption by students (for self-assessment or review prior to midterm or final examinations)</td>
<td>No change in number of adoptees within the civil engineering technology (CVT) department</td>
<td>Increased number of students within CVT department adopting use of the site</td>
<td>Adoption by some students in other engineering departments within the same institution</td>
<td>Adoption by some students in other institutions offering engineering technology programs</td>
<td>Widespread adoption possible</td>
</tr>
<tr>
<td>Adoption by instructors (for take-home assignments)</td>
<td>No change in number of adoptees within the civil engineering technology (CVT) department</td>
<td>Increased number of instructors within CVT department adopting use of the site</td>
<td>Adoption by some instructors in other engineering departments within the same institution</td>
<td>Adoption by some instructors in other institutions offering engineering technology programs</td>
<td>Widespread adoption possible</td>
</tr>
</tbody>
</table>

Notes:

1. Most questions will include detailed graphics, which may contain standard symbols (pinned connection, smooth roller, distributed loads, etc.). There will be dimensions, loads, etc., that overlay the graphic (providing

² 68 is the average web system usability score
individualized inputs for each user). It is not expected that screen-readers will be able to interpret all, or even much, of this information.

1.7.2. User Survey

The first ten questions are slightly modified from the System Usability Scale (Brooke, 2013) standard set of question, modified here to be more specific to the current project. Survey respondents should give answer these questions on a scale of 1 to 5, where 1 indicates “strongly disagree” and 5 indications “strongly agree.”

Questions 11 to 15 are follow-up questions and depend upon potential usability issues identified in the first ten questions and are short answer.

1. (Student) I would find Qwizm useful for self-review and revision before a test or an exam.
   (Instructor) I would demonstrate Qwizm in class to show that it is an effective equivalent to textbook problems and exercises.
2. I found the system unnecessarily complex (n.b., This survey question refers to the Qwizm site itself, and not to the relevance or difficulty of the engineering course-related content).
3. I thought the Qwizm site was intuitive to navigate and use.
4. I would need technical support to be able to utilize this site.
5. I found the various functions in this system well-integrated.
6. I thought there was confusing inconsistency in the system.
7. I would expect most users to learn the system quickly.
8. I found Qwizm cumbersome to use.
9. I felt confident using the system very quickly.
10. I needed a lot of trial and error to figure out how to use the site.

11. If, in survey question 2 above, your answer was 4 or 5 (i.e., “agree” or “strongly agree”), please describe what you found unnecessarily complex and, if possible, how you would remedy this complexity. Three or four sentences should be sufficient.
12. If, in survey question 4 above, your answer was 4 or 5 (i.e., “agree” or “strongly agree”), for which parts of the site, or for which operations, would you need technical support?
13. If, in survey question 6 above, your answer was 4 or 5 (i.e., “agree” or “strongly agree”), where did you find inconsistency?
14. If, in survey question 8 above, your answer was 4 or 5 (i.e., “agree” or “strongly agree”), can you make any suggestions how to make the system less cumbersome? Three or four sentences should be sufficient.
15. If, in survey question 10 above, your answer was 4 or 5 (i.e., “agree” or “strongly agree”), which parts of the site were initially confusing to you? Three or four sentences should be sufficient.
1.8. Life of the project beyond capstone

The primary purpose of this capstone is to prepare a clean and efficient codebase that will be generated by a future desktop quiz and question authoring application (provisionally named Qwizard). Qwizard will generate the html and JavaScript for multiple-answer numerical questions. It will also compile selections of these questions into quizzes that can be distributed in a single .html file.

An ancillary purpose of this capstone is to modernize my outdated version of this project, taking advantage of more recent technologies (local storage, CSS3, ...) and facilitating a more efficient development of questions. This will be useful in the interim, until my final desktop application is complete.

My hope is that Qwizard will be a useful tool for those teachers and instructors wishing to create such questions and quizzes without requiring a programming background and knowledge of JavaScript. Former colleagues have expressed an interest in using such an application in their own courses.

I have been happily retired for the last 15 months so I do not intend that this project will lead to a job. There is a possibility of a future business, not in the sale of the application itself but in the distribution of question libraries I hope to author; in that case, I would not opensource the questions themselves. Currently, I feel that such a business approach is unlikely.

Questions and quizzes that I create using the results of the capstone will be hosted at http://qwizm.org/ for continued use by all who are interested. Each significant addition of material, or module, will be accessible from Merlot (http://merlot.org/) to reach a wider audience.

I plan on making the source code for the final application, and possibly for a subset of developed questions, available on Github—enabling maintenance if/when I am no longer able to perform this maintenance myself.

2. Competitor Review

The difficulty here is finding competitors to review. I have identified only one direct competitor\(^3\) for online quizzes that require multiple accurate numerical answers for each question or problem: this competitor is the Formulas plug-in for the Moodle LMS (Formulas Question Type, 2014) described in 2.3 below.

In addition to the Formulas plug-in, I am choosing to review: a standard LMS implementation (using Blackboard as an example in this case) of a single numerical

\(^3\) Which serves to reinforce the need for a product such as Qwizm
answer question type; and the standalone (i.e., not contained within an LMS) Hot Potatoes application that, whilst not offering randomized inputs to numerical answer questions, does export quizzes as a single HTML file for hosting online or direct email distribution to students.

2.1 Blackboard: Calculated Formula Questions

2.1.1 Description
“Calculated Formula questions present students with a question that requires them to make a calculation and respond with a numeric answer. The numbers in the question change with each student and are pulled from a range that you set.” (Calculated Formula Questions, 2018)

2.1.2 Pros
- Existing authoring tool.
- Can be configured to check units.
- Online marking and writing to gradebook.

2.1.3 Cons
- Only a single answer is permitted.
- Access limited to Blackboard platform users.
- Blackboard is not free
- Closed source.
- No instant feedback; quiz must be submitted for online marking before quiz-taker may view whether they are correct or not, which is not so useful for formative learning.
- No randomized inputs displayed on any graphic included in the question. Instead, variable names ($x_1, x_2, ...$) on the graphic must be matched with values provided in the question statement. (See example below, taken from a sample quiz created using the Formulas plug-in for Moodle.)

2.2 Hot Potatoes

2.2.1 Description

2.2.2 Pros
- Variety of question types.
- Freeware.
• Immediate feedback.
• After a long hiatus, a beta version of Hot Potatoes 7 has been released. This version generates HTML5 code.
• Able to export quiz as a single HTML file.
• Able to export quiz as a SCORM 1.2 package (SCORM 1.2, 2018).

2.2.3 Cons
• No numerical question type of any kind, making this unsuitable for the type of problems suited to Qwizm.
• Closed source.

2.3 Moodle: Formulas Question Type Plug-in:

2.3.1 Description:
“... a question type plugin for Moodle with random [input] values and multiple answer fields” (Formulas Question Type, 2014)

2.3.2 Pros:
• Existing authoring tool.
• Multiple answers or parts for each question as required.
• Can be configured to check units.
• Online marking and writing to gradebook.

2.3.3 Cons:
• Access limited to Moodle platform users.
• Requires course registration to view quizzes.
• Requires a special plugin that your Moodle admin may not be willing to install.
• Only 224 downloads (Moodle Plugins, 2019) so limited usage.
• Due to retirement, no current maintainer.
• ‘Instant’ feedback for each part has to be ‘gamed’: After each answer, you must mark the whole question and then look for a checkmark. This requires that the question be set up for unlimited penalty-free attempts.
• No randomized inputs displayed on any graphic included in the question. Instead, variable names ($x_1, x_2, ...$) on the graphic must be matched with values provided in the question statement. (See example below, taken from a sample quiz created using the Formulas plug-in for Moodle.)
Example quiz using Formulas plugin on Moodle. Login with username ‘learner’ and password ‘Le@rner1’ at http://eduk8r.org/moodle/mod/quiz/attempt.php?attempt=84&page=2
(Morgan, eduk8r.org/moodle, 2018)
Table 2 Feature Comparison for Qwizm and alternatives.

<table>
<thead>
<tr>
<th>Features</th>
<th>Blackboard Calculated Formula Question</th>
<th>Hot Potatoes</th>
<th>Formulas Plug-In for Moodle</th>
<th>Qwizm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical Question Type</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Multiple Answer</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Authoring Tool</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗¹</td>
</tr>
<tr>
<td>Dynamic Overlay of Randomized Inputs on Graphics</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Feedback During Question Entries</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Fully Responsive Site</td>
<td>Unknown²</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Automatic Gradebook Entry</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗³</td>
</tr>
<tr>
<td>Direct Navigation to Quiz (1 or 2 mouse clicks)</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Free</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Open Source</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

¹ A primary goal of Qwizm is to prepare a concise codebase to facilitate the creation of a future authoring tool.
² Blackboard quizzes are not publicly accessible.
³ Note that automatic gradebook entry is not always an advantage. Certainly, it does save time but the instructor does not get to view each student’s written work. The suggested procedure for submitting Qwizm is for the student to print out the summary page (showing the entries and marks earned for each question) and attach a ‘professional’ or neat copy of their written work/calculations to the printed summary. The instructor can quickly scan the work and then manually enter marks in the gradebook, frequently taking less than one minute per student.
3. Technology Requirements / Resources / Material

Qwizm will run completely in the browser so there aren’t any practical alternative software technologies to HTML, CSS and JavaScript. Some JavaScript libraries may be incorporated to facilitate typesetting of mathematics.

Technologies

HTML5
- **Description:** Mark-up language for display of website content
- **Related courses:**
  - DGMD E-20  Developing Interactive Media
  - DGMD E-27  Modular and Mobile Front-End Design
- **Alternative technologies:** No alternative for web-site display
- **Reason for choice:** N/A
- **How used:** For display of site content

CSS3
- **Description:** Styling language for websites.
- **Related courses:**
  - DGMD E-20  Developing Interactive Media
  - DGMD E-27  Modular and Mobile Front-End Design
- **Alternative technologies:** No alternative for web-site display
- **Reason for choice:** N/A
- **How used:** To make the site attractive, intuitive to navigate, and responsive so it is practical to use on most sizes of device.

JavaScript
- **Description:** Programming language that is interpreted by the browser.
- **Related course:** CSCI E-3  Web Programming/JavaScript
- **Alternative technologies:** Server-side technologies, such as PHP or NodeJS, could be used to replace most, if not all, of the JavaScript.
- **Reason for choice:** Going forward, after the completion of this capstone, I shall be building a quiz- and question-authoring tool; it will generate the Qwizm code (a primary purpose of this Qwizm capstone is to determine a codebase suitable for generation by this future authoring tool). The authoring tool will enable question creation and quiz compilation by those with no programming knowledge, and will export a quiz as a single HTML file containing all the questions and (encoded) images. Each quiz can then be easily distributed to students as an email attachment, or hosted on an institution’s shared drive. (My former teaching colleagues had nowhere they could, or the skills or inclination to, host server-side code.) A quiz, packaged
in a single file and viewable in all standards-compliant browsers, will be a valuable course asset.

- **How used:** JavaScript will perform the bulk of the work. A few examples are:
  - The `Window.localStorage` property will be used to maintain the state (i.e., the answers already entered) of a quiz between browser sessions.
  - Creation of a linear congruential generator\(^4\) to generate ‘random’ question inputs within a certain question-specific range: For example, to generate a pipe-length between 500m and 1500m in length, with lengths being in increments of 25m (i.e., generate one of 500, 525, 550, ..., 1500).
  - Create a cypher to encode and decode numbers in variables so that they are not (readily) accessible using browser developer tools.
  - Control the CSS for displaying and hiding separate views (login, introduction, question 1, ..., summary view)
  - Mark submitted answers correct or incorrect, to the specified precision.

**Hosting**

Qwizm ([http://qwizm.org/](http://qwizm.org/)) will remain hosted by the current hosting platform [https://www.siteground.com/](https://www.siteground.com/) who provide good service at a reasonable cost. This is shared hosting and not suitable for large traffic (the current plan can accommodate approximately 25,000 visits monthly) but, in the event that this becomes an issue, upgrading to 100,000 visits is simple.

---

\(^4\) [https://getbootstrap.com/docs/4.0/layout/overview/](https://getbootstrap.com/docs/4.0/layout/overview/)
4. Developer Manual

The figure below shows the directory and file layout for http://qwizm.org/. Quizzes are sorted by course subject and linked directly from the homepage. The homepage is a simple html file with some text describing potential uses for the site but is essentially a table of contents for all quizzes.

![Diagram of directory and file layout for http://qwizm.org](image)

**Figure 3 Directory and file layout for http://qwizm.org**

Each individual quiz, e.g. Force Components, is accessed through a link on the Qwizm homepage following a path such as http://qwizm.org/statics/components/components1/index.html (as can be seen in the figure above). There may be more than one quiz that relates to force components; in that case, each force component quiz will be in a separate quiz directory and have a separate link from the site homepage.
Files qwizm.js and utils.js are common to all quizzes: qwizm.js extracts data from quiz.js (specific to each quiz) and builds the quiz site; and utils.js contains functions used by the quiz.js. These utility functions perform the following:

- random number generation used for setting individual input values for each question,
- conversion of floats to specified number of significant digits, converting to one more than the specified number if the first non-zero digit is a 1, e.g., convert to 2.34 (3 significant digits) and to 1.234 (4 significant digits),
- string formatting to significant digits (i.e., 12.00 instead of 12 if four significant digits are required),
- conversion of radians to degrees and vice versa,
- marking of input answers correct or incorrect, to the specified number of significant digits,
- encoding and decoding of variables containing correct calculated answers to mask these values if students use browser developer tools to investigate the correct solution,
- ...

Files qwizm.scss and qwizm.css, controlling the responsive nature of the site, are common to all quizzes. Breakpoints will be the same as those used by Bootstrap although the Bootstrap float-based grid system will be avoided, if possible, in favor of the newer CSS3 Grid and Flexbox layouts. Secondary styles (statics.scss and statics.css) are defined at the subject level and generally serve only to set different color schemes for different subjects.

When the page index.html has loaded, Qwizm will check localStorage to see if there exists a theQuiz object for the current quiz (each quiz has a unique ID). If the current theQuiz object does exist, it means that the user has logged into the quiz before; in this case, the theQuiz object will be loaded into memory and the user presented with her the view she last used on her previous visit. (Views are Login, Info, Clear, Q1, Q2, ..., Summary). If the current theQuiz object does not exist, the user will be provided with a Login view before being taken to the Info view.

Each quiz is a single page; each of the views is present, if not visible, all the time. For example, navigating from Question 2 to Question 3 is a matter of turning off the display of Question 2 and turning on the display of Question 3. Navigation is performed by clicking on buttons in the page footer; transitions are quick since there is no return trip to the server, just the application of a couple of CSS properties. Once loaded into the browser, the quiz should have the responsive feel of a native application.

---

5 [https://getbootstrap.com/docs/4.0/layout/overview/](https://getbootstrap.com/docs/4.0/layout/overview/)
5. User Manual

Login View

The first time you visit a quiz with a particular device, you are presented with the Login View.

Enter your name (using only alphabetical characters) and your ID (only integers).

Then, press the Log In button to proceed to the Information View for your quiz.

Information View

This is where you read the quiz author’s specific instructions for this quiz.

Pay especial attention to instructions relating to the significant digits required both for the answers and for your interim calculations.
Question View

There is a separate view for each question in the quiz.

Enter the answer for each question part, then verify your answer by pressing the ‘Check’ button. You will get immediate feedback about the correctness of your answer and the marks earned.

If you answer is incorrect, you may try again. You have unlimited, penalty-free, attempts.

Numerical entries are maintained between browser sessions.
Summary View

At any time, you may click the Summary button to assess your progress through the quiz so far.

If this quiz is used as a take-home assignment by your instructor, you may have to print out this view and submit it with your written work.

Clear View

The Clear View allows you to reset the quiz and remove all your previous entries and to return you to the Login View.

You may want to clear your quiz in order to enter a new, fictitious, ID number; this will allow you to practice questions with different input values.
6. Work Plan and Milestones

Table 3: Project Milestones

<table>
<thead>
<tr>
<th>Milestone</th>
<th>Completion Date</th>
<th>Deliverable(s)</th>
</tr>
</thead>
</table>
| Views complete                     | January 27th, 2020  | • Login, Info, Clear, Question and Summary Views complete (excluding content) and navigable  
                                 |                     | • Site responsive                                                             |
| Question functionality complete    | February 24th, 2020 | • Question object built with all key/value pairs including ranges and increments for each input variable, question statement, user answers required derived from input variables, static image, locations/placement of input variables on top of image  
                                 |                     | • Each question comprises three blocks: question statement, question image and required answers. These must flow naturally for different screen or browser window sizes.  
                                 |                     | • Input variables overlaid on the static image should maintain correct placement for different screen, browser sizes. |
| Quiz functionality complete        | March 9th, 2020     | • Integration of questions into the quiz site  
                                 |                     | • Implementation of summary table and integration into quiz                   |
| Sample content complete            | April 6th, 2020     | • Create a simple set of questions and incorporate into a quiz. This content will be non-engineering specific. These sample questions will be used for testing and debugging the question logic. The testing and debugging is not anticipated to take very long since incremental testing will be employed during the building of the question functionality. |
7. References


