Lance Desrouleaux, Troy Martin, Adam Donner & Reva Baumann

Web 330

Assignment 2.3

30 April 2019

**Programming Penguins**

Team Captain: Lance Desrouleaux

Our Repository: https://github.com/inlighten1/Web-330-Group-Project

Team Bio:

Programming Penguins is a group of Web Development students with a tenacity for creating something awesome. We develop intuitive web solutions that help our clients excel. As experts in the field, our design services come from years of experience, a track record of success, and a drive to always create something bigger and better. The most effective solutions that we provide contain a simple interface, clean elements and don’t overwhelm the user. We want to create something that presents our client's ideas in a way that’s efficient, compatible and entertaining.

**Meet Our Team**



**Lance Desrouleaux**

Lance has always had a passion for all thing technology and the digital arts. He attended a Magnet school where he studied Computer Programming with a focus in Java and C++. As a hobby, Lance taught himself the basics, HTML, and CSS. Previously he worked for a manufacturing company doing software development using Python. Currently, he works doing tech support for a Web Development company that specializes in web apps for e-commerce websites who are on the Shopify, 3dcart, and Yahoo platforms. Before Lance was working in e-commerce, he was working in the AV Field. In his free time, Lance performs as a DJ at local events and produces music. He passionate about traveling and has lived in several countries including Thailand, the Philippines, and Peru. Currently in school to get a formal degree in full stack web development so that he can open doors to more opportunities to help people increase their web presence.

**Troy Martin**

Troy Martin is a software developer with over twenty years of experience creating desktop and web-based applications. With a varied technology background, Troy specializes in building complete web based solutions using Microsoft .Net C#, HTML, Javascript and Microsoft SQL Server.

Troy has an Associates of Applied Science degree in Architectural Drafting and started programming writing custom functions for AutoCAD computer-aided drafting stations in a scripting language called AutoLISP. Switching fields, Troy worked for several years designing network installations and setting up file servers for point of sales systems.

Troy went back to programming and developing web applications after being introduced to the server side language, ColdFusion which he used to solve a reporting issue for a coworker. After several years developing ColdFusion applications Troy switched to developing Microsoft .Net web applications in C# and has been using the Microsoft stack for the last fifteen years. With a passion for learning new technologies, Troy is currently learning Angular, TypeScript, and other modern web-based technologies and is researching machine learning. Troy enjoys travel, reading, running, and taking long walks with his wife and dog.

**Adam Donner**

Adam Donner has spent the past 15 years of his career in the software industry in both the engineering and sales/marketing divisions. After graduating from the University of Nebraska, Adam started his career as a software engineer for a company that develops printers. Adam spent four years as an engineer for that company before moving into a regional manager of both the engineers and software salespeople. After four years in the business, Adam left the company to become a regional sales director for a large software company that develops voice and imaging technology.

As a regional manager Adam was responsible for one-quarter of the US in his vertical market and as Adam was accustomed to traveling with his previous employer traveling every week was not a problem until his twin boys, Owen and Alex were born. After five years Adam decided that he was going to stop traveling and became a Director of Sales and Marketing for a local company where he and his team is responsible for selling and developing e-commerce websites for key accounts in Nebraska and Iowa. In his spare time, Adam likes to camp, mountain biking, and hiking.

**Reva Baumann** 

Reva has spent the last few years as a Web Development hobbyist, creating HTML & CSS websites for friends and family. She started coding as a teenager, using simple Wordpress and personal blogging websites. Although she lacked formal training, she spent much of her time building rudimentary websites and projects in her free time, learning and studying as much as she could. Much of her time was spent with her cousin, Dylan, who’s a professional web developer and challenged her to learn on her own.

Reva has a background in Art with a bachelor’s degree she obtained from UNO. After college, she worked in Sales, working in Healthcare Staffing and a large e-commerce company. When Reva was working in sales, she worked directly with Web Developers to integrate the technology she sold to the client’s website. This rekindled her love of website design and creation. Reva enrolled in Web Development classes at Bellevue University to obtain her degree and a formal background in development. Her specialty is in Web Design, leaning on her experience in art to balance website aesthetics. In Reva’s free time, she enjoys reading, playing video games and working on her art.

**Application Description:**

Single page application using HTML, CSS, JavaScript, Bootstrap 4, jQuery, and KnockoutJS.

In the browser using JavaScript maintain a list of questions, answers to the questions and the user’s answers. There are several ways we can structure the classes used in JavaScript to maintain the state of the quiz and user answers. For example.

A quiz object with the collection of questions, answers, and method to calculate the summary.

* Quiz object
  + Method to calculate rank.
  + Method to calculate the percentage.
  + Question collection
    - Question object
      * Question text.
      * Collection of answers.
      * Current user’s answer.
      * The correct answer to the question.

Sample JSON structure:

|  |
| --- |
| let quiz = {  getRank : function(){  ...  },  getScore: function(){  ...  },  questions: [  {  question: "This is the question",  answers: [  {  identifier: "A",  answer: "Answer text"  }  ],  usersAnswer: "A",  correctAnswer: "A"  }  ] } |

**Opening section**

* Display the title of the application.
* Capture name of the user?
* Start the quiz.

**Quiz section**

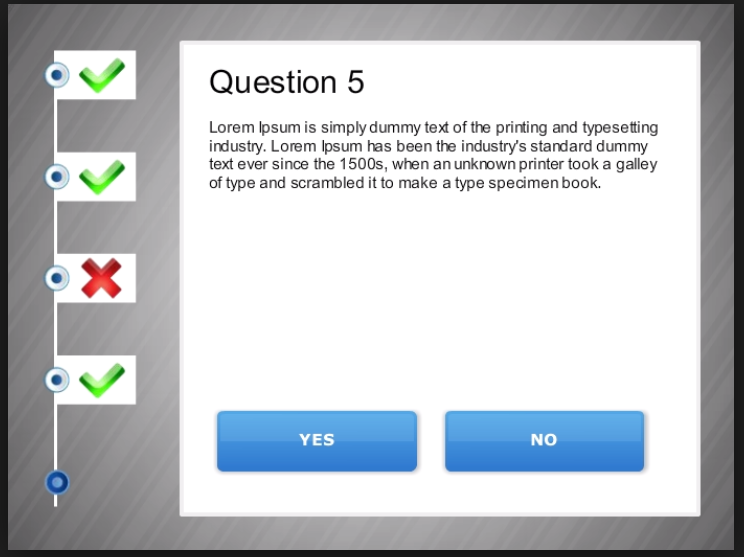
* Display the current question and its answers.
  + Display the user’s progress in the quiz
* Record the user's answer and move to the next question.
* The user should be able to return to the previous question.
  + When returning to a previously answered question the user will have the ability to change the selected answer.

**Summary section**

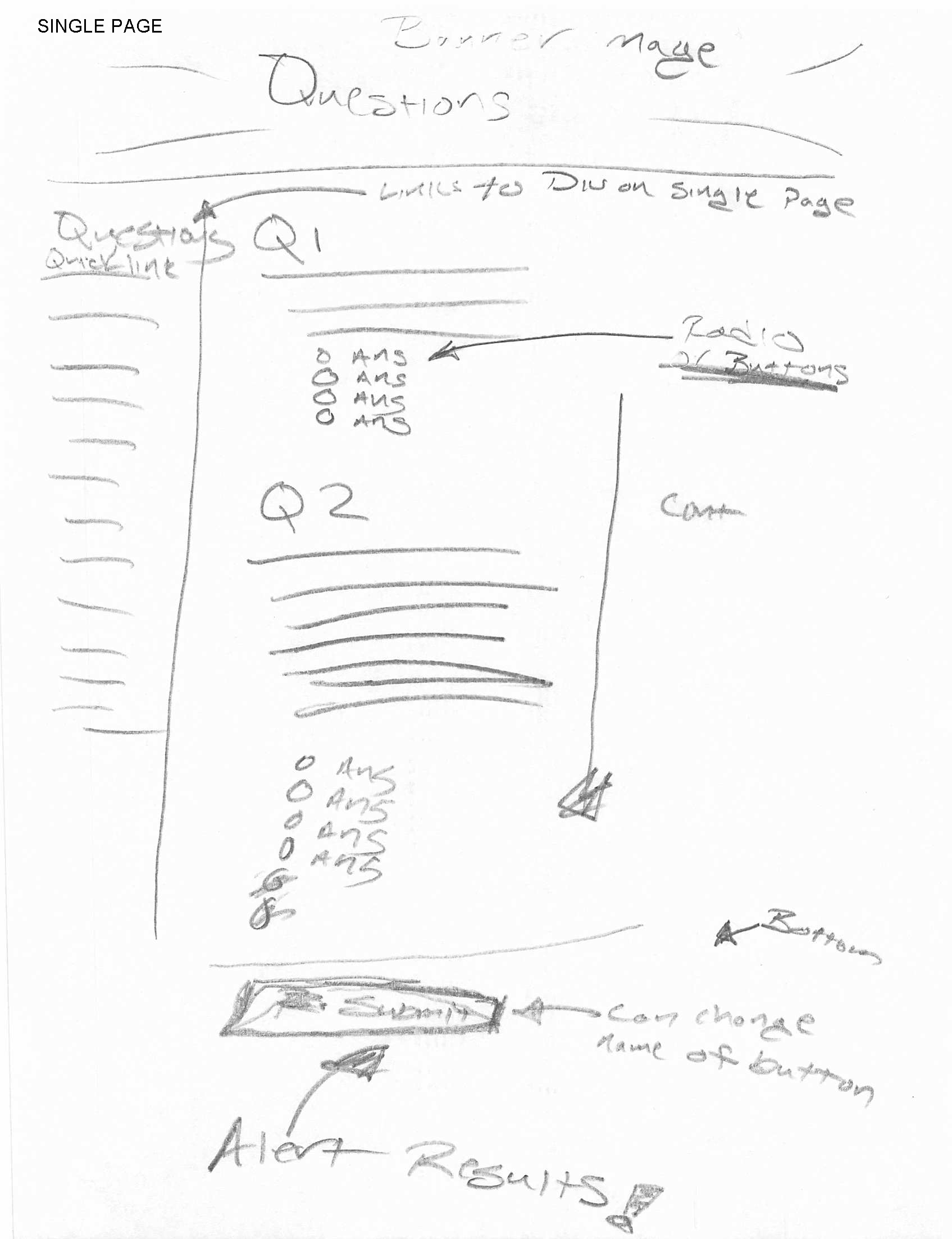
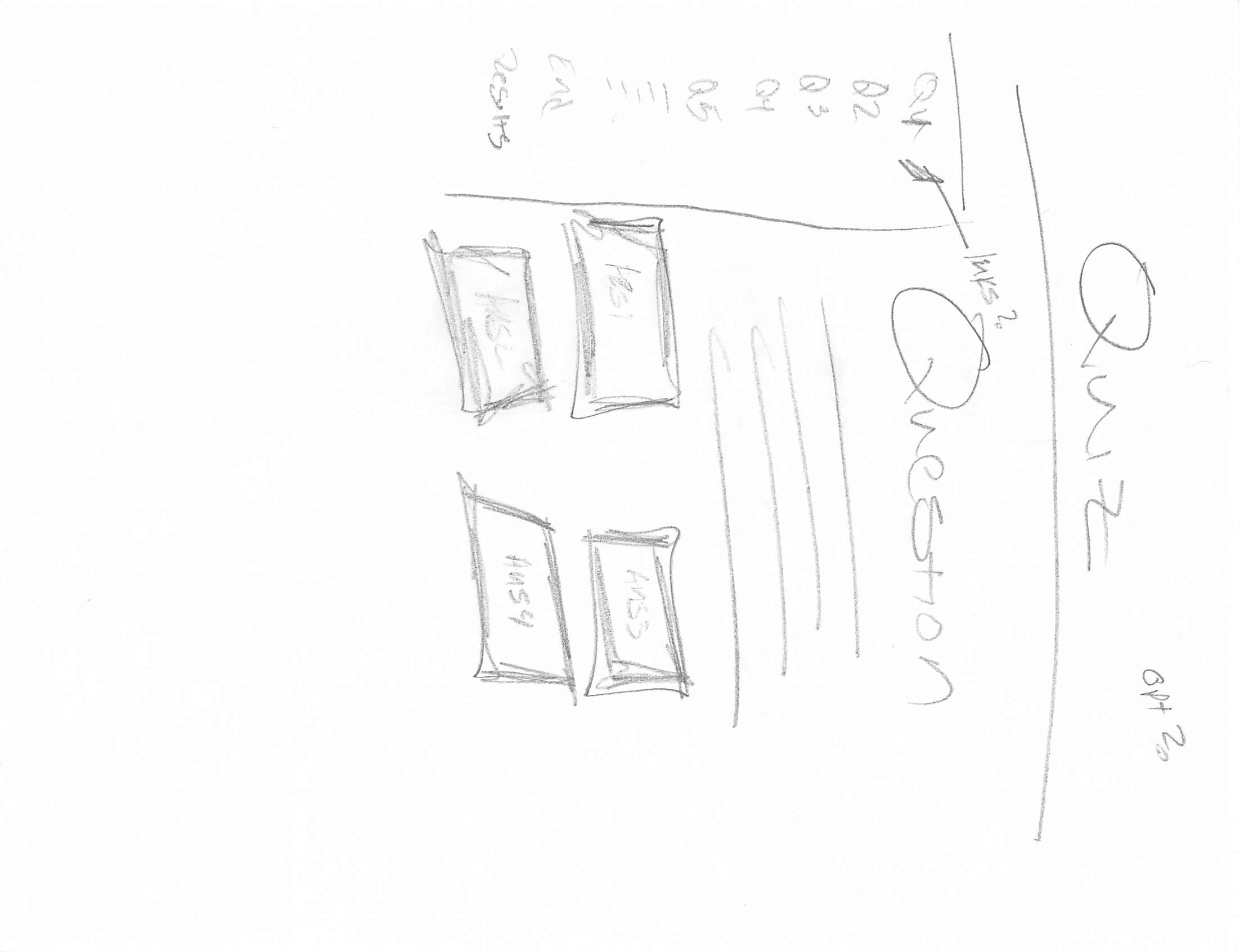
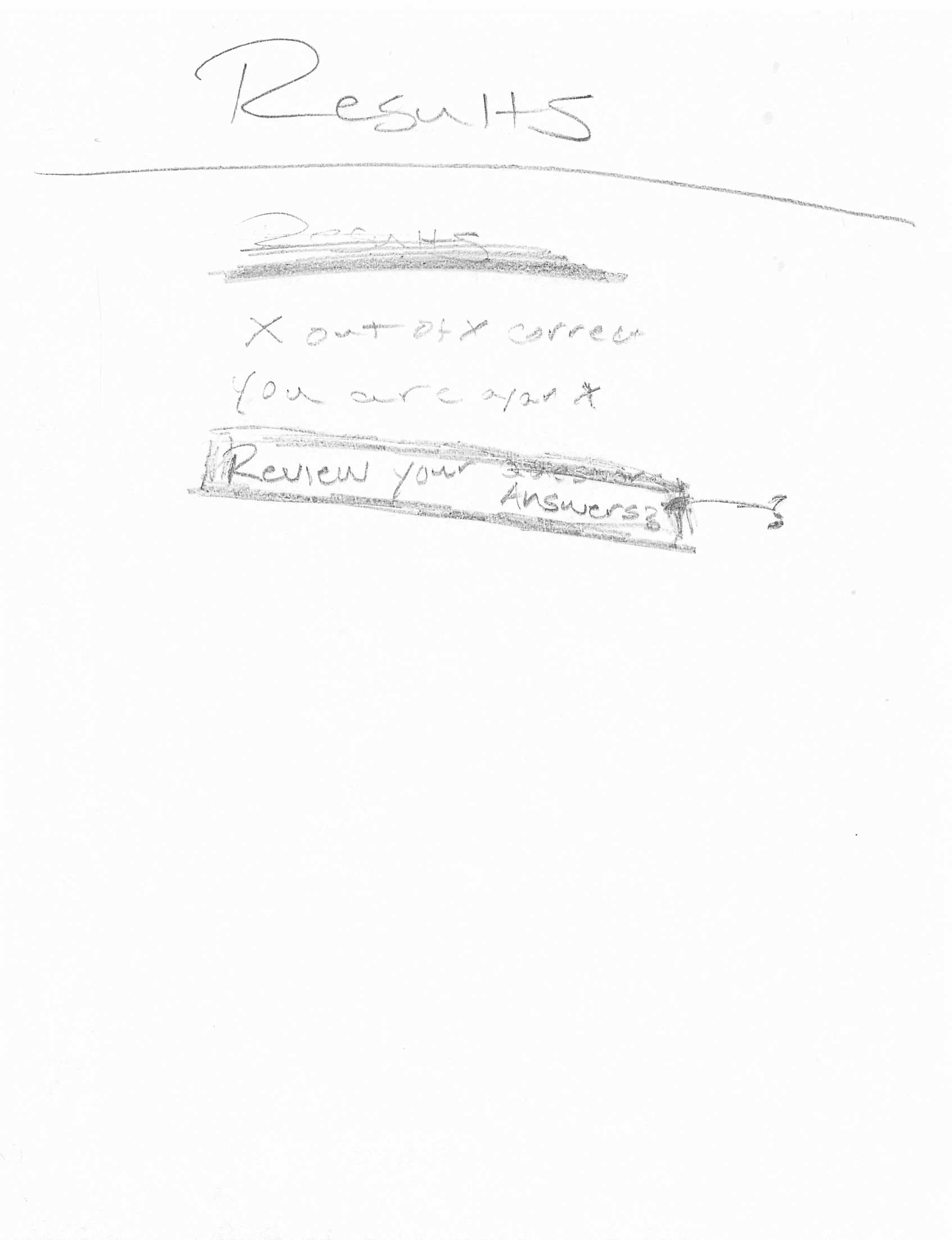
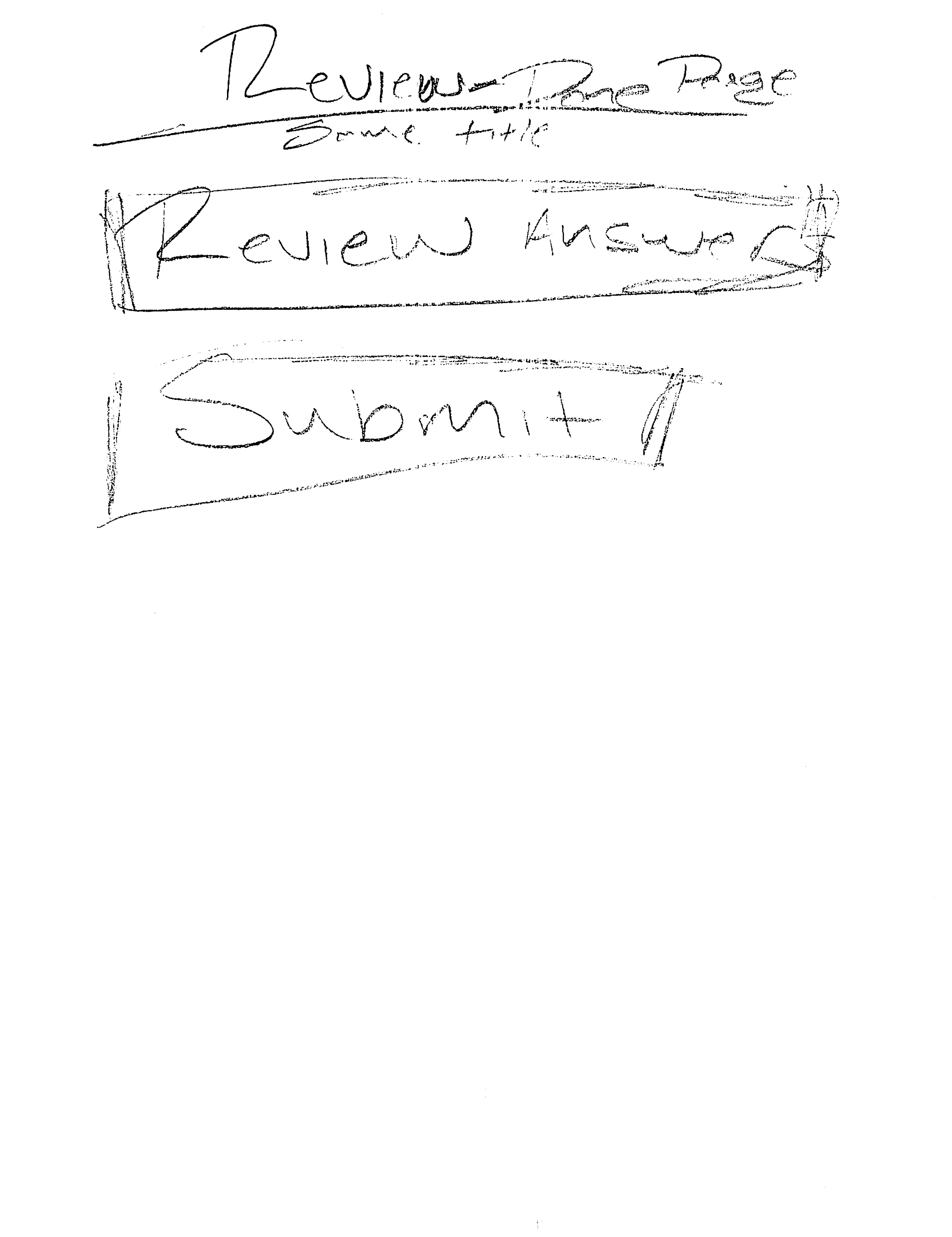
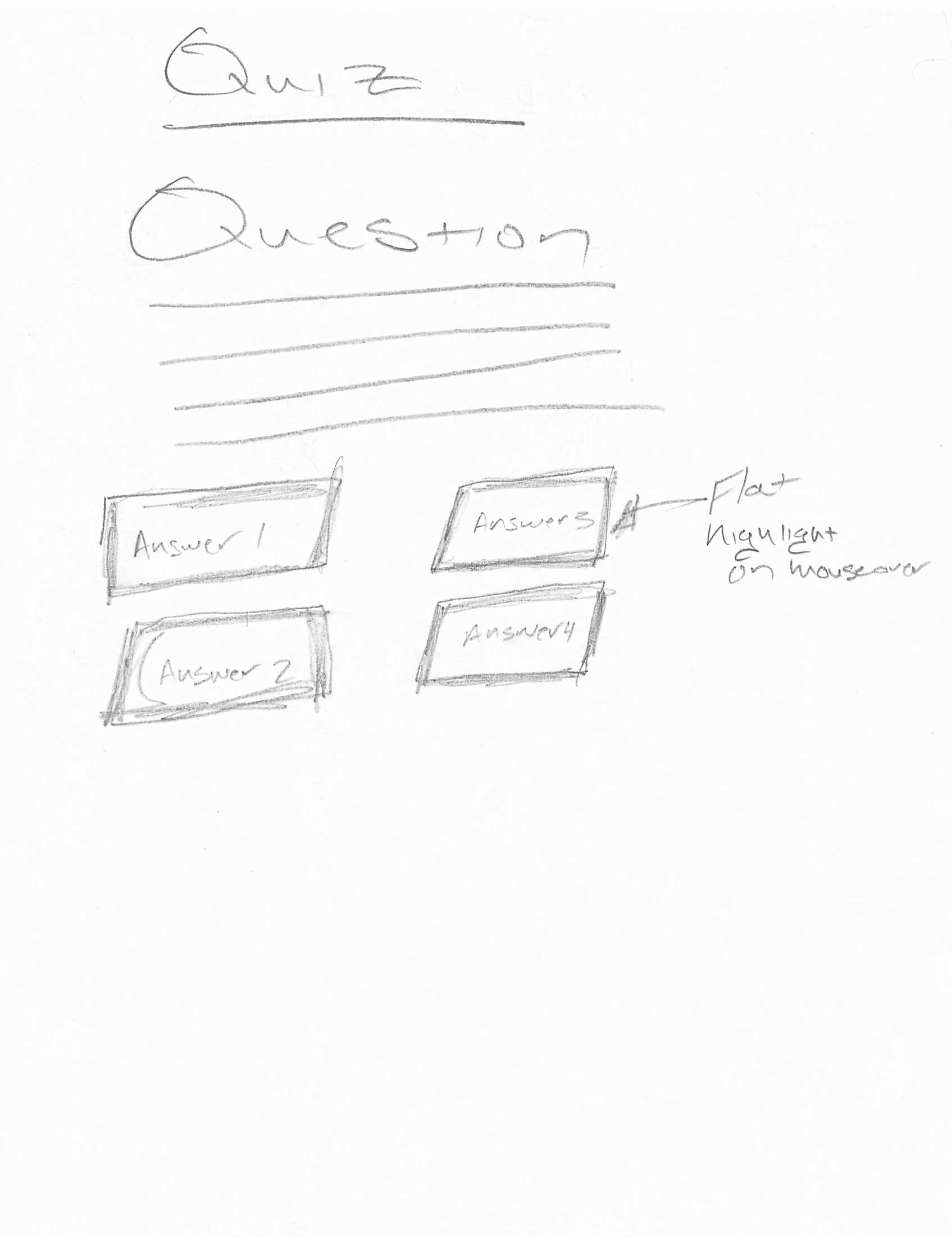
* After the last question is answered display the summary section.
  + Display rank, count the questions in the collection that the user’s answer does not match the questions answer.
    - Expert 8 - 10 correct
    - Novice 6 - 8 correct
    - Beginner less than 6 correct
  + Display percentage score, count the questions in the collection that the user’s answer does not match the questions answer.
    - (*n* / 10) \* 100
    - Round the answer to a whole percentage?
  + Display all the questions, correct answer and user answer.

Layout ideas:

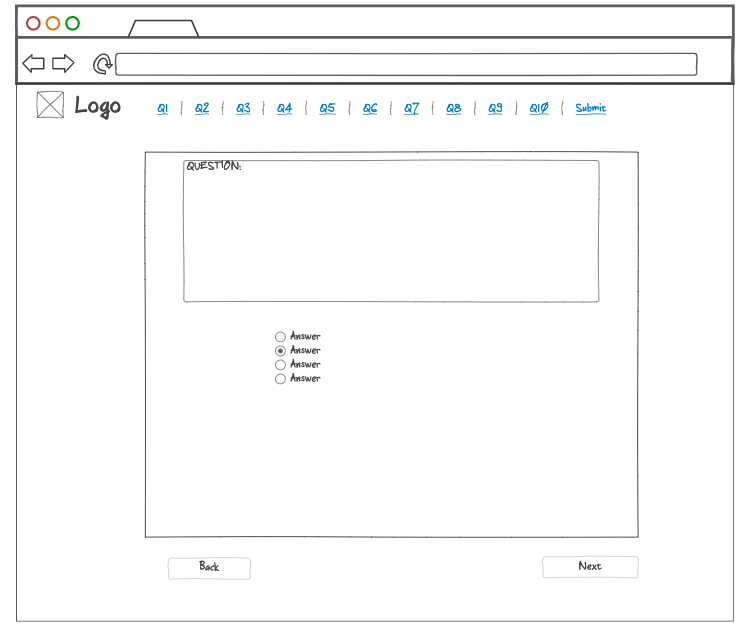
Show status as the user progresses through the quiz.

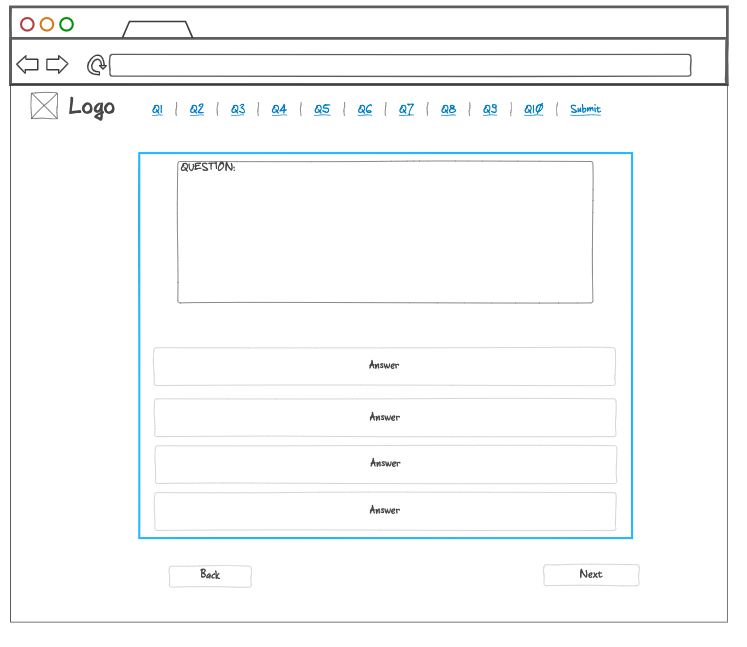


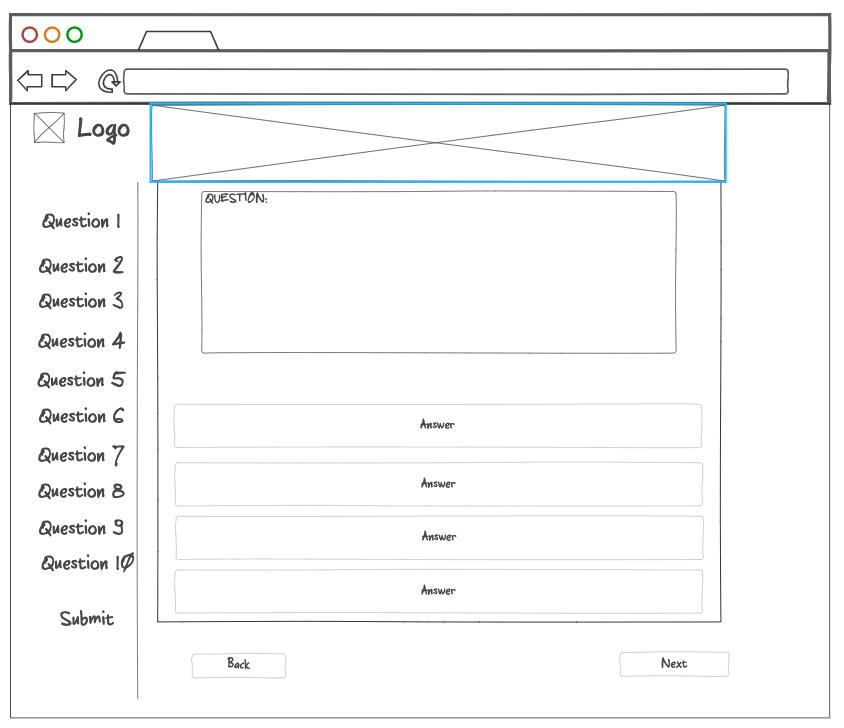
|  |  |
| --- | --- |
| Show the rank and score as the user progresses | Ways to display the question and answers |
|  |  |

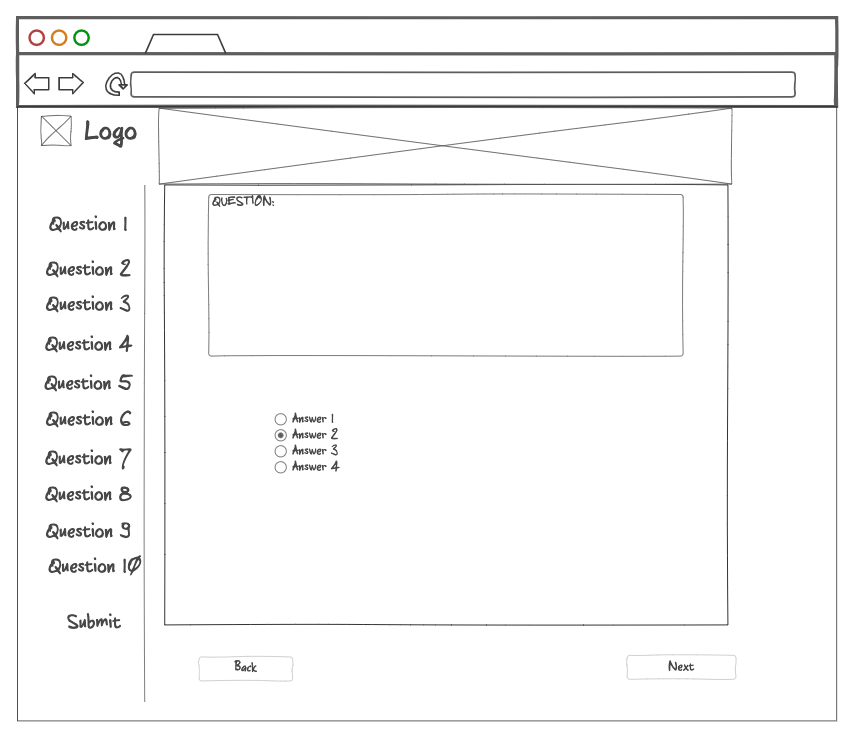


Wireframes







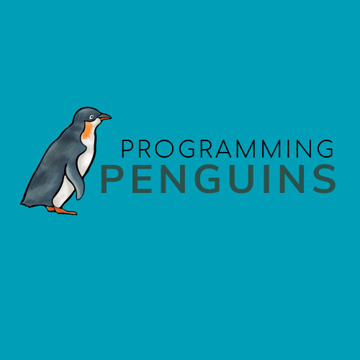




**Logo Design**





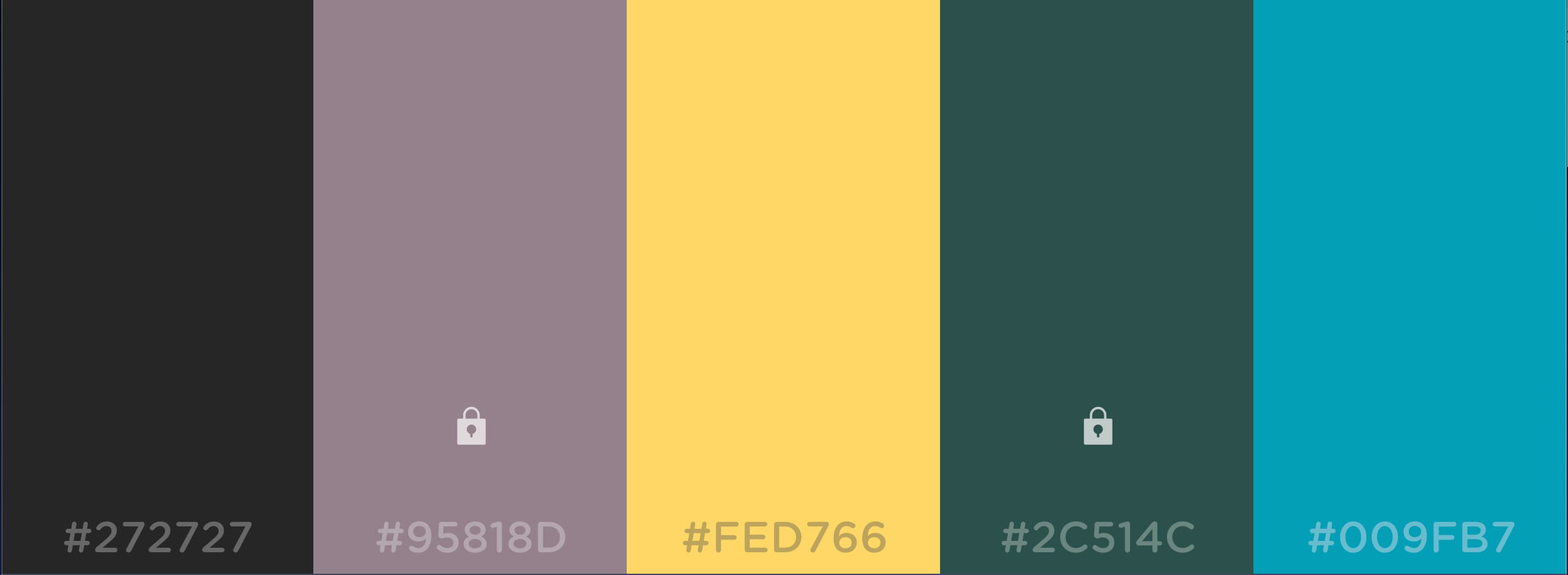
 



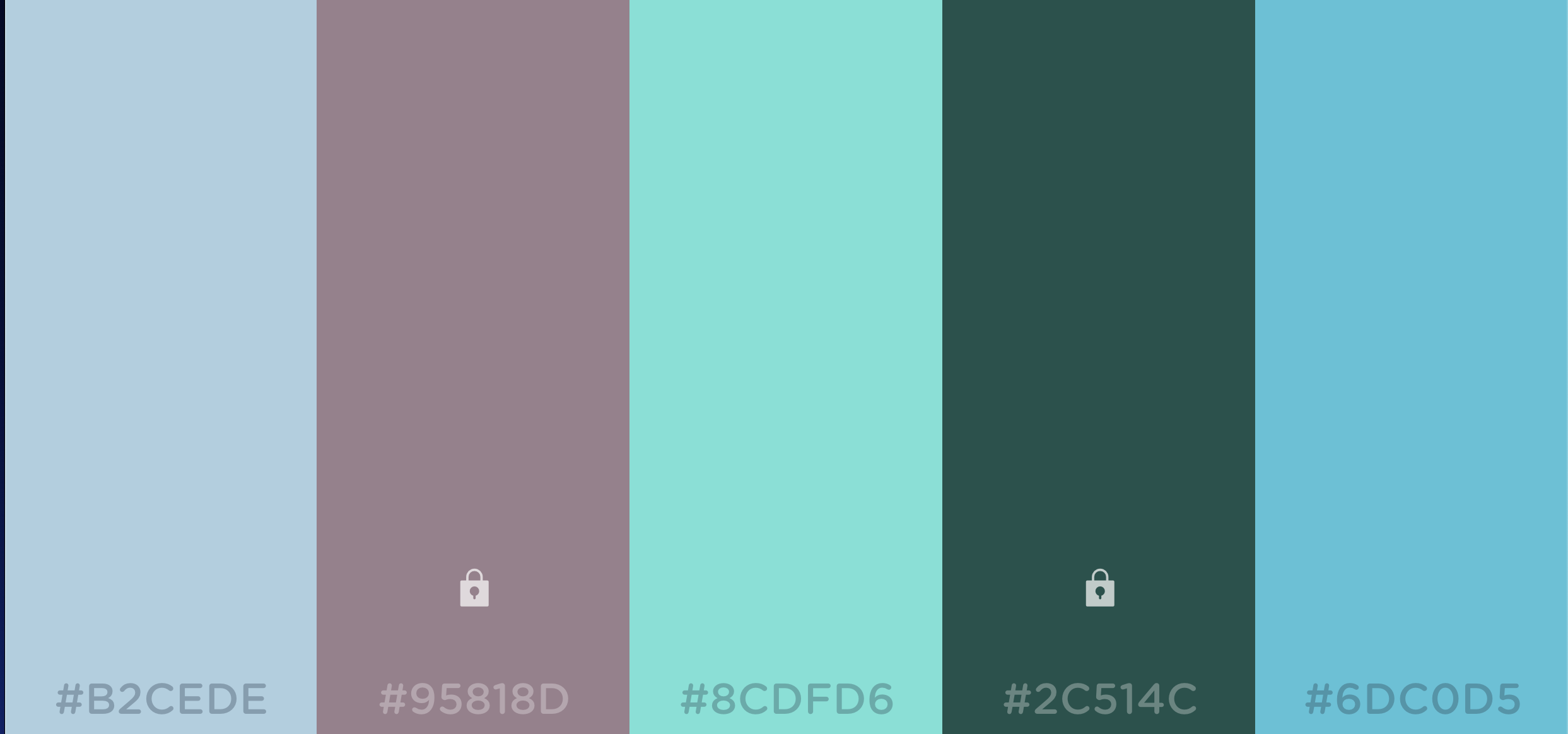
**Logo:**

We wanted to incorporate fun color schemes that are bright and eye catching, but also simple. We wanted the penguin graphic to be more realistic than cartoony, and added the colors of the penguin to offset the color schemes we liked. We went with a simple, bold typeface that was alluring and easy to read.

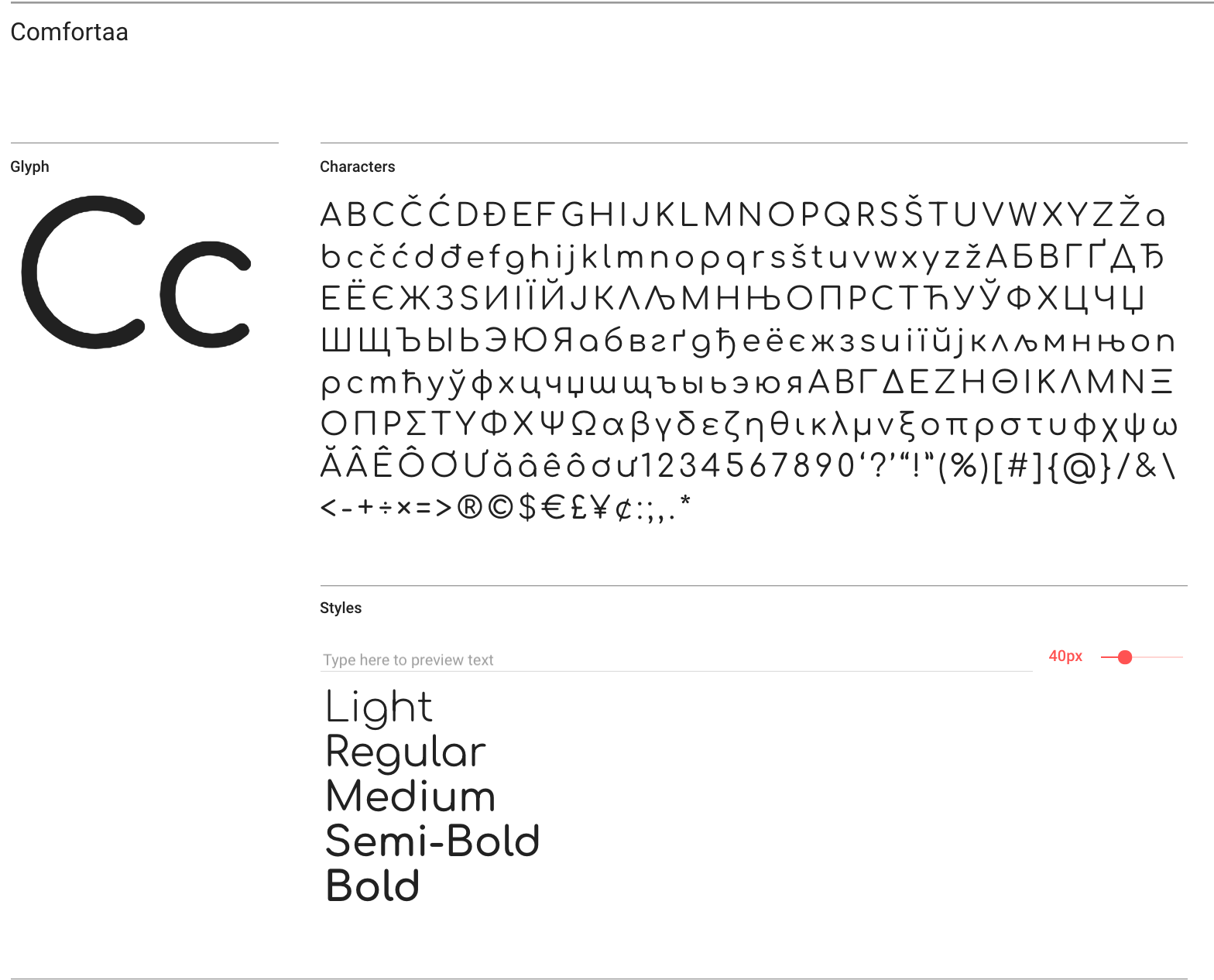
**Color Schemes:** We wanted to integrate something fun, playful and cohesive in color. We liked cool tones, with pops of bright energy.

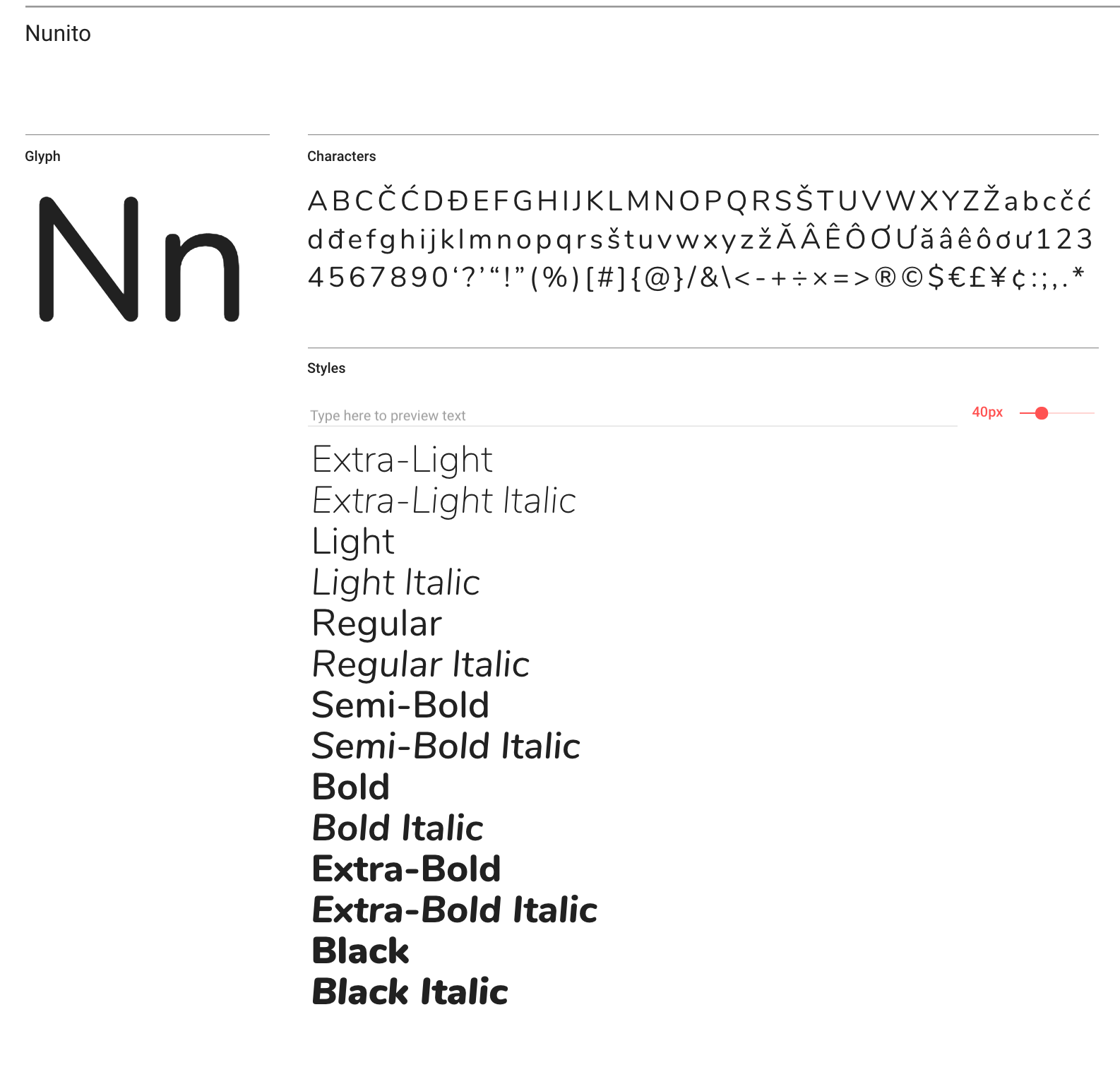






**Font Schemes:** We wanted to incorporate fun fonts, that are both professional and playful. We used Nunito and Comfortaa to combine these aesthetics, as well as create a cohesive color scheme and brand.





Quiz Questions

Is this a proper for statement?

for (var counter = 0; counter < 10; counter++) {

var chance = randomNumber();

counter = 6;

}

1. True
2. **false**

An array is an order collection of values. What are those values called.

1. Index.
2. **Element**
3. Object
4. Property

The value of an evaluated express in a switch statement can be a string.

1. **True**
2. false

What is the term used for the space between the curly braces of a function?

1. The function parameters.
2. The function arguments.
3. **The function body.**
4. The function whitespace.

What is the name of the function on the console object used to output values to the screen?

1. console.draw
2. **console.log**
3. console.output
4. console.writeLine

Which is an example of a constructor prototype?

1. Song.constructor.toString = function(){

return `${this.title} performed by ${this.artist}`;

}

1. Song.toString = function(){

return `${this.title} performed by ${this.artist}`;

}

1. this.toString = function(){

return `${this.title} performed by ${this.artist}`;

}

1. **Song.prototype.toString = function(){**

**return `${this.title} performed by ${this.artist}`;**

**}**

Inside what HTML element, do we put JavaScript?

1. **<script>**
2. <javascript>
3. <js>
4. <scriptjs>

How do you write a message in an alert box?

1. **alert("message")**
2. msgBox("message")
3. msg("message")
4. alertBox("message")

How do you create a function in JavaScript?

1. **function myFunction()**
2. function:myFunction()
3. function = myFunction()

Which of these is the correct syntax?

1. function clearAll() {

document.getElementsByClassName("independence")[0].style.visibility = 'hidden';

};

1. function clearAll() {

getElementsByClassName("independence")[0].style.visibility = 'hidden';

};

1. function clearAll() {

document.getByClassName("independence")[0].style.visibility = 'hidden';

1. function clearAll() {

document.("independence")[0].style.visibility = 'hidden';

}

Which of these is the correct syntax?

1. $(btn).on("click", function(){}
2. $(#btn).on(click, function(){}
3. $("#btn").on("click", function(){}
4. (#btn").on("click", function(){}

Which of these is the correct syntax?

1. <button type="button" onclick= "document.getElementById('.furniture').innerHTML = new Date()">
2. <button type="button" onclick= "getElementById(‘#furniture').innerHTML = new Date()">
3. <button type="button" onclick= "getElementById('furniture').innerHTML = new Date()">
4. <button type="button" onclick= "document.getElementById('furniture').innerHTML = new Date()">