

## CMPT 276 Assignment 2

Due: Oct 11, 11:59:59pm.

### Overview

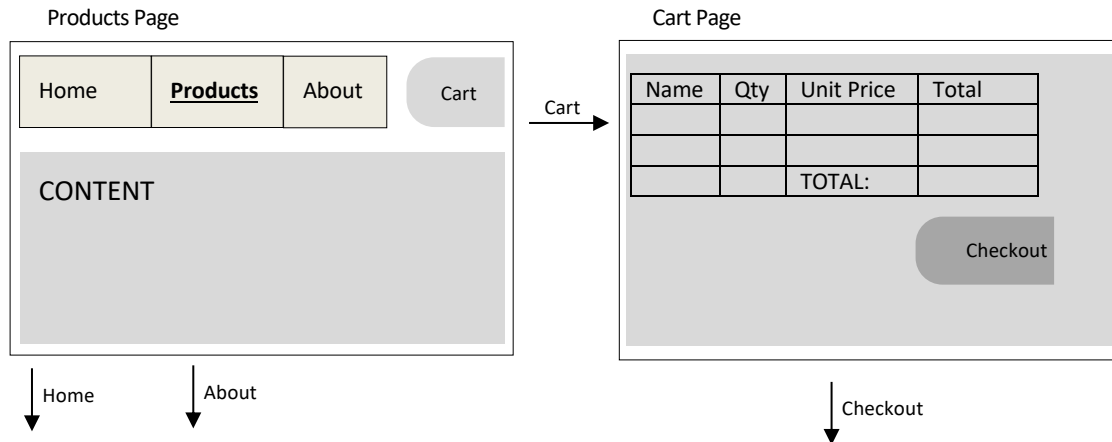
A Tokimon is a rare creature found in the remote areas of Korea. For the most part, they resemble rabbits and the only form of communication they have with us is the ability to say their own name. Tokimons come in all shapes and sizes, have special abilities, and tend to fight with each other on occasion. Each Tokimon could have the ability to fly, fight, throw fire, spray water, electrify, and freeze other Tokimons; their ability is measured by an integer between 0 and 100. Because they are such primitive creatures but have such extraordinary abilities, they need Tokimon trainers. The trainer's job is to collect Tokimons and use them accordingly. We would like to write an application to track the known Tokimons in existence.

- Database: you should create a database table (called "Tokimon") to collect information about a series of creatures. The minimum attributes for the Tokimon are name, weight, height, fly, fight, fire, water, electric, frozen (or ice), total (total is just the sum of their abilities), and name of trainer (**as a string**). You may add any other Tokimon attributes as you see fit.
- Features: your app should have the following features:
  1. The ability to add new Tokimon (with corresponding attributes).
  2. The ability to change attributes of any of the Tokimons.
  3. The ability to delete any of the Tokimons.
  4. The ability to display information about a specific Tokimon. A link (Links) should be provided from the Tokimon info page that links to more info.
  5. The ability to display all Tokimons currently in the database (including any other information you collect).

In addition, you may also add or display any other attributes you wish. Please note that the requirement stated above is a minimum list, you may add to them as you see fit. Please be creative, part of your grade will be based on effort, usability, and creativity (see 'Usability and Effort').

## UI Mockup

Begin with a UI mockup of your app. You may use any tool that you desire (including by hand). You do not have to hand this in, but you will have to include this as part of your documentation in subsequent assignments. The UI mockup should contain your different views and how they are linked. As a small example, a UI mockup for part of a bakery (buying) website could look like the following:



## Usability and Effort:

Once again, a small portion of the grade will be dedicated to usability and effort,

Usability is learnability and likeability. Once again, existence of bugs such as lack of error checking, dead links, poor workflow (navigation) may result in a lower usability score. Other forms of usability include the likability/creativity of the display pages.

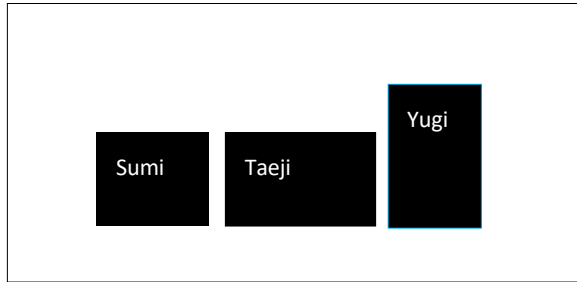
For the Effort mark, add at least two other features to this application; this will be a chance for you to play around with the message passing framework of NodeJS and help us to think about our group project.

An example of an extra feature would be to display different types of Tokimon differently. One idea is to display Tokimon according to their weight and height. Let's say the database looks something like:

| Tokimon Database |       |        |        |      |       |     |
|------------------|-------|--------|--------|------|-------|-----|
|                  | name  | weight | height | fire | water | ... |
| 1                | Sumi  | 150    | 30     | 100  | 0     |     |
| 2                | Taeji | 200    | 30     | 0    |       |     |
| 3                | Yugi  | 120    | 50     | 20   | 70    |     |

The display page *could* look something like this (notice Sumi and Taeji have the same height but Taeji is proportionally heavier, Yugi weighs less and is taller than both Sumi and Taeji):

Display Page



Again, if you are unsure whether your idea is an acceptable feature, please see either the TA or myself. Please clearly indicate your extra features in the 'README.md' file found in the root of your application.

### **Marking Scheme:**

2 marks: correct submission of webapp and git link – We must be able to find your working app at the webapp link, and should be able to clone your project at the git link (must be either github or gitlab link, we cannot clone the heroku git link).

3 marks each: features 1-3, 5 (described earlier)

4 marks: feature 4 including the properly implemented links between them

2 marks: proper calculation of 'total' Tokimon ability ~~and the 'level' of the trainer~~

5 marks: Usability and Effort.

### **Submission**

Deploy your work to Heroku and submit the URL of your webapp and git link to the coursys server. <https://courses.cs.sfu.ca/>

**THE END**