

## **Artifact 1: <veekun-pokedex2>**

### **Overview (1 - 2 paragraphs)**

The Domain • Pokedex: A digital encyclopedia designed to give information about the Pokemon in the world contained in its database • Pokemon data is interesting and has a flexible level of complexity • Pokemon community is very robust (PokeAPI, Veekun, online Pokedexes, random excel spreadsheets posted to forums) • Each Pokemon game has a unique Pokedex with its own variants. • We stuck with the National Pokedex to keep the database focussed. • Business needs? Hobbyists, competitive players, people who want to know details about specific Pokemon

**Technology (1 - 2 paragraphs)** java and db browser is used in the project. Java for the code db browser for the database

### **Resources**

Provide link(s) to the project.

### **Screenshots**

Provide an appropriate number of screenshots of the project in order to give the reader a “sense” of the project running. Screenshots ARE NOT of the code but rather the project running.

## Example Artifact 1: Attack of the Killer Snowmen (Escaping Reality Project)

### Overview

Northpole has been invaded by killer snowmen. It's your job to save the magical ice cubes and return them home in order to vanquish the invaders. This is a first person game where you are placed amidst a forest surrounded by the dormant invaders waiting for their time to strike. You have 60 seconds to collect 5 magical ice cubes and return them to the house in the forest. If you are successful, the invading killer snowmen will be sent back from where they came, otherwise you will be their next victim.

### Technology

Attack of the Killer Snowmen is built using AFrame.io, a web framework for creating 3D / VR / AR experiences. Because AFrame.io is a web framework it has elements of HTML and JavaScript. Minimal static HTML was used to set up the environment. Most of the HTML used to render the experiences was dynamically generated using JavaScripts. Classes were used to model the snowmen and ice cubes. 3D models from Sketchfab.com were used to represent the trees, snowmen, ice cubes and the house.

### Resources

<https://github.com/rcastro2/AttackOfTheKillerSnowmen>

### Screenshots



## Example Artifact 2: SpongeBob Adventures (We Code You Play Project)

### Overview

Help SpongeBob get to work at the Krabby Patty in order to feed his friend Patrick. Beware of Squidward, Mrs. Puff and the troublesome jellyfishes looking to stop you from getting to work. If SpongeBob manages to get to work, he must be careful of Plankton who is looking to cause havoc at the Krabby Patty as SpongeBob attempts to feed Patrik.

The first stage of the game SpongeBob must drive himself and Patrick to the Krabby Patty. Avoid Squidward, Mrs. Puff and the troublesome jellyfishes as each provides their own unique challenge to prevent SpongeBob from getting to work. The second stage involves assembling Krabby patties in the right order so that he can feed Patrick. Don't take too long as Patrick is extremely hungry. Watch out for Plankton who is looking to slow you down as you assemble the Krabby patties. Plankton is probably trying to steal the secret recipe for the Krabby patties.

### Technology

SpongeBob Adventures is built using Python, PyGame and gamelib.py. Various objects such as Images and Animations were created using classes found in gamelib.py. The logic of the program involves using separate "game loops" to represent each "screen" in the game. There was extensive setup required prior to each game loop. Both stages of the game used lists to store the various moving pieces. For loops were used to traverse these lists while in the game loops in order determine the result of the interaction with the objects in the list. Various Spritesheets and graphics were obtained through the internet.

### Resources

<https://replit.com/@rennecastro/Spongebob-Adventures>

### Screenshots



## Example Artifact 3: How to Make a Meal (Web Development)

### Overview

Ever stare into your refrigerator wondering what you can eat? How to Make a Meal empowers the user to search for recipes based on a specific ingredient. A list of potential recipes are displayed allowing the user to select a particular recipe. Upon selecting the desired recipe, an image of the meal, required ingredients and steps for the recipe are displayed. YouTube is also displayed if available.

### Technology

Web pages were developed using HTML and CSS. JavaScript was used to provide the functionality of retrieving information and dynamically displaying to the page. JQuery was used to retrieve JSON information from [TheMealDB](#) through API calls. Functions and classes were used to abstract the logic of the program into manageable components.

### Resources

<https://replit.com/@rennecastro/2122WDT4Lesson551CSolution>

### Screenshots

This screenshot shows the main search interface. At the top, there's a blue header with the title "How To Make A Meal". Below it is a search bar with the placeholder "Search for a meal...". Underneath the search bar is a list of meal names: Cream Cheese Tart, New York cheesecake, Three-cheese souffles, Honey Yogurt Cheesecake, Peanut Butter Cheesecake, Chicken Fajita Mac and Cheese, Grilled Mac and Cheese Sandwich, and Fruit and Cream Cheese Breakfast Pastries. At the bottom right of the search bar is a blue "Search" button.

This screenshot shows the details for a selected recipe. The title "Cream Cheese Tart" is at the top. Below it is a "Ingredients" section with a table:

250g	Flour
125g	Butter
1	Egg
Pinch	Salt
300g	Cheese
100ml milk	Milk
3	Eggs
100g	Parmesan Cheese
350g	Plum tomatoes

Below the ingredients is a detailed description of the recipe steps. On the right side of the page is a small thumbnail image of a tart and a large black rectangular area where a video player would be located, indicated by a play button icon.