# CIS 218 Lab 07: Team Project (80 points, 5 bonus points possible)

Please leave the lab questions/instructions/rubrics/etc. in place. Just paste your screenshots and code below my instructions.

Purpose/knowledge/skills: This week, you’ll have a chance to exercise your vue.js skills to create a project of your own choosing and design.

## Task 1: Vue.js project (80 points, 5 bonus points possible)

Create a vue.js page of your own design. It’s fine to start with lab files from a previous week or built-in project templates. The work for this project comes in creating content and adapting the existing structures to your needs.

A list of sample topics is included below. You are welcome to create something else so long as your project meets all requirements.

## Sample project topics

A. Gallery of pet images – your own, or images found on the internet

B. A news site that allows users to click to different articles without changing the entire web page

C. Redesign one page of the SPSCC web site

This is you reimagining how one page might be designed! You are **not** responsible for recreating functionality or even all the content of any particular page.

D. Interactive web site for children that uses JavaScript events to reward them using the mouse to explore the page/site. One approach is mouseover events: pictures that wiggle, rockets that launch, …, but other designs are acceptable.

E. An input form for user information to register at a hospital.

## Possible resources for the project

* Royalty-free stock images from <https://www.pexels.com/>
* <https://www.w3schools.com/css/css3_images.asp>
* <https://www.w3schools.com/CSSref/css3_pr_filter.asp>
* <https://www.w3schools.com/css/css3_transitions.asp>
* <https://www.w3schools.com/css/css3_animations.asp>
* <https://vuejs.org/guide/extras/animation.html>

## Project requirements

Your project must use:

1.1. Vue.js

1.2. Animations and/or transitions

1.3. At least one of the following:

3.1. Conditional rendering

3.2. Forms and input

3.3. Looped rendering (v-for)

3.4. Single-file components (SFCs)

## Steps to complete the task

1. Identify the web site you want to make.

2. Sketch out a rough diagram of how you want the site to look and work. This gives you a target to work towards. There is no penalty if the final design does not match your initial work!

3. Code the site together or as a team.

3.1. You do not need to cite content sources for this project. Copy and reuse G-rated content freely.

3.2. Vue.js code must be your own, or cited if you find non-trivial code online.

4. Create a short write-up about your project, including:

4.1. At least three screenshots

4.2. A one- or two-paragraph (total) description of the features of your site

4.3. A brief description of how it meets project requirements 1.1, 1.2, and 1.3, about one sentence per item.  
 A template for the write-up page is provided at the end of this document

5. Capture the project for submission. Create a .ZIP file with:

5.1. The source files for your web site, including all files needed to launch your project

It’s okay to link to online libraries such as vue.js. You do not need to capture those locally.

5.2. The write-up you created in step 4.

## Rubric

* Site works: 20 points
* Use vue.js: 5 points
* Animations and/or transitions: 5 points
* Additional tech: 10 points
* Project write-up: 20 points
* Creativity and extra effort: 20 points
* Teamwork: +5 bonus points

## WEEK 6 PROJECT WRITE-UP

Team member name(s): **Jahan Polatova, Clayton Tupper**

Project name:

**Children’s Computer Learning App**

Screenshot #1:

A computer screen with text and a cartoon of a yellow dog on a computer monitor

Description automatically generated

Screenshot #2:

A screen shot of a computer

Description automatically generated

Screenshot #3:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Briefly describe your site and its major features. This should be about one paragraph.

**Our site is a web-based children’s learning app. We created a few different modules using vue sfcs(single-file-components) so the user can play simple mini-games and learn to navigate a keyboard and mouse. The landing page and games are each their own component and navigated with links and routing.**

Briefly, how does your site meet requirements? This should be about one sentence per item.

* 1. Vue.js:

**We used vue’s component-based approach to create the app, and incorporated many of vue’s features like conditional rendering with v-if directives.**

* 1. Animations and/or transitions:

**After one of the minigames (mouse practice), a new element is shown on the page using both transitions and animations.**

* 1. At least one of the following: conditional rendering, vis, looped rendering (v-for), single-file components (SFCs):

**We used several of them – each of our games is defined as a single-file-component, we use looped rendering with v-for and conditional rendering with v-if in the mouse game.**