

Module #5 Plan | CSE 310 – Applied Programming

Name	Date	Teacher
Nathan Luckock		Bro McGary

Software Description

A simple drawing and painting web app built with TypeScript. Users can draw on a canvas with different brush sizes and colors. The app will include a color picker, brush size slider, clear button, undo functionality, and the ability to save drawings as images. It will have a clean, simple interface that works well on desktop and mobile browsers.

Module

Mark an **X** next to the module you are planning

Module	Language
Cloud Databases	Java
Data Analysis	Kotlin
Game Framework	R
GIS Mapping	Erlang
Mobile App	JavaScript
Networking	C#
Web Apps	x TypeScript x
Language – C++	Rust
SQL Relational Databases	Choose Your Own Adventure

Create a Schedule

Create a detailed schedule using the table below to complete your selected module during this Sprint. Include details such as what (task), when (time), where (location), and duration. You should also include time to work on your team project. You are expected to spend 16 hours every Sprint working on your individual module, team project, and other activities. Time spent on this individual module should be at least 10 hours.

	First Week	Second Week
Monday	Set up TypeScript project, create canvas and basic drawing (2 hours)	Add color picker and brush size controls (1 hour)
Tuesday	Implement clear and undo functionality (2 hours)	Add save/download image feature (1 hour)

	First Week	Second Week
Wednesday	Polish UI and add mobile touch support (2 hours)	Testing and bug fixes (1 hour)
Thursday	Add extra features like different brush shapes (1 hour)	Create video demo and documentation (2 hours)
Friday	Team project work (1 hour)	Final UI polish and features (1 hour)
Saturday	Write README and documentation (1 hour)	Final testing and GitHub push (1 hour)

Identify Risks

Identify at least two risks that you feel will make it difficult to succeed in this module. Identify an action plan to overcome each of these risks.

	Risk	Action Plan
1	Canvas drawing might not work smoothly on touch devices	Test touch events early, use pointer events API for better mobile support
2	TypeScript setup and configuration could be confusing	Follow a simple TypeScript tutorial first, start with basic setup and add complexity later