Software Requirements Specification

for

A7: Sprite Editor

**Version 1.0 approved**

**Prepared by Team Ctrl Alt Elite**

**Ashlyn Hansen, Emma Luk, Lindsey Peterson, Ricardo Baeza**

**CS 3505**

**November 01, 2023**

**Table of Contents**

[**1. Introduction 1**](#_heading=h.1fob9te)

[1.1 Purpose 1](#_heading=h.3znysh7)

[1.2 Document Conventions 1](#_heading=)

[1.3 Intended Audience and Reading Suggestions 1](#_heading=)

[1.4 Product Scope 1](#_heading=)

[1.5 References 1](#_heading=)

[**2. Overall Description 1**](#_heading=)

[2.1 Product Perspective 1](#_heading=h.2s8eyo1)

[2.2 Product Functions 1](#_heading=)

[2.3 User Classes and Characteristics 2](#_heading=)

[2.4 Operating Environment 2](#_heading=)

[2.5 Design and Implementation Constraints 2](#_heading=)

[2.6 User Documentation 2](#_heading=)

[2.7 Assumptions and Dependencies 3](#_heading=)

[**3. External Interface Requirements 3**](#_heading=)

[3.1 User Interface 3](#_heading=h.2jxsxqh)

[3.2 Hardware Interfaces 4](#_heading=)

[3.3 Software Interfaces 4](#_heading=)

[3.4 Communications Interfaces 4](#_heading=)

[**4. System Features 4**](#_heading=)

[4.1 Sprite Size Setting 4](#_heading=)

[4.2 Add/Delete Frames 4](#_heading=)

[4.3 Animation Preview 5](#_heading=)

[4.4 Pen 5](#_heading=h.q524yxoccpsy)

[4.5 Shape Tool 5](#_heading=h.koe5yip3rgz1)

[4.6 Eraser 6](#_heading=h.x6a8tlr8ls32)

[4.7 Color Picker 6](#_heading=h.d7fyj7wjau16)

[4.8 Fill Bucket 6](#_heading=h.7qs5jw4ck0wj)

[4.9 Save and load 7](#_heading=h.a51ba07zqjmf)

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Develop a Qt-based **Sprite Editor**. A sprite is a small image used in gaming that is rendered at different locations on the screen, most sprites have animation cycles associated with them.

For example, a sequence of drawings might show the sprite walking, or exploding, or powering-up, etc.

## Document Conventions

**Bold:** Reserved for the name of the **project** and names of **widgets** and **classes** within our document**.**

Underline: Reserved for names of features such as Pen Tool and Selection Tool

## Intended Audience and Reading Suggestions

This document is intended for the developers of this product and project managers (i.e. David Johnson and the rest of the cs3505 course instructors/ teaching assistants.

For an overall sense of the **Sprite Editor** product and goals we suggest reading section 2, and referring to the overview of the application’s functionality and layout in section 4 for greater detail.

## Product Scope

The purpose of this software is to allow a user to edit different frames in a sprite. This will allow for easier creation of software that uses sprites and for storing multiple frames of a sprite in one file.

## References

*“Qt Modules.” Qt Documentation, doc.qt.io/qt.html. Accessed 5 Nov. 2023.*

# Overall Description

## Product Perspective

This **Sprite Editor** is a new, self-contained product for David Johnson, regarding his CS3505 - Software Practice II Assignment Seven.

## Product Functions

The product will allow users to edit different frames of a sprite using different pen tools, including free-hand, straight line, fill bucket, rectangle, and circle shape tools. There will also be an eraser tool for users to erase mistakes. They will also be able to open existing sprites and to save these sprites to their computer.

## User Classes and Characteristics

CS3505 Professor:

Characteristics

* High level of expertise.
* All access admin privileges.
* Very high education level

Needs:

* The project must encompass fundamental features expected in a **Sprite Editor**, including tools like a pen tool, shape tool, color picker, and the capability to save and create files. This evaluation aims to ensure that the developers' proficiency aligns with the learning outcomes of the CS3505 course.

CS3505 Teaching Assistants:

Characteristics

* Varying levels of expertise
* All access admin privileges
* Varying levels of education

Needs:

* Similar to professor needs, the project must encompass fundamental features expected in a **Sprite Editor**, including tools like a pen tool, shape tool, color picker, and the capability to save and create files.
* Code must be readable, well-documented and modular for evaluation.
* **Sprite Editor** must have the ability to run on a variety of operating systems.

## Operating Environment

We are creating the application in QT, and we will save each version to github for sharing with the group. Both Windows and Mac operating systems are present in our group, so it will have to function with both of these.

## Design and Implementation Constraints

Need to use Qt Creator Widget Application and should maintain model-view structure. We are limited to about a week to create the assignment and we are anticipating using many existing Qt classes, so we will be limited either to existing API or our ability/willingness to create new methods and/or classes.

## User Documentation

A help section within the application will be accessible to provide users with guidance on utilizing the **Sprite Editor** and its range of tools.

## Assumptions and Dependencies

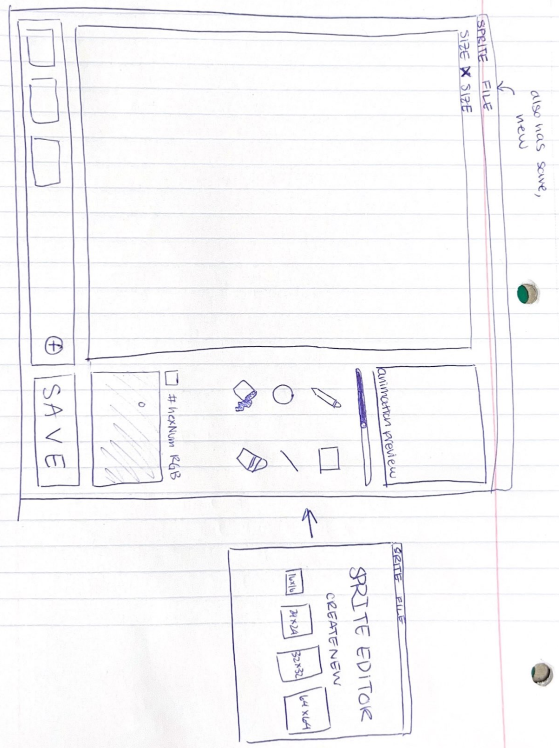
Third-Party Components:

* QT Native API widgets(including but not limited to):
  + **QIODevice**
  + **QFile**
  + **QDataStream**
* Github Version Control
  + will be used during development to manage and control multiple contributor’s code.

# External Interface Requirements

## User Interface

We will keep every element of the software in one window. Tool buttons will be in a widget down the right side of the screen, frames will be selectable in a scroll-bar along the bottom, and drawing/editing space will take up the majority of the center of the screen. A preview of the animation (cycle through different frames) will be in the top right corner.



## Hardware Interfaces

The bulk of the software will be controlled by a mouse as the user clicks the different buttons or clicks and drags to draw/edit their sprites.

## Software Interfaces

Operating Systems:

Windows versions( 10+):

* Connection to file manager for opening and creating files and directories within Windows.

MacOS versions (10.15+):

* Connectionto finder for opening and creating files and directories within MacOS

## Communications Interfaces

**Sprite Editor** will only be storing files locally and thus require no communication interfaces.

# System Features

## Sprite Size Setting

4.1.1 Description and Priority

The ability to set the size in pixels of the sprite. Options are 16x16, 24x24, 32x32, or 48x48.

Priority: High

4.1.2 Stimulus/Response Sequences

Can be done when creating a new sprite document. All frames after this will be of the same size.

4.1.3 Functional Requirements

REQ-1: Must be a “start” page for creating a new sprite

REQ-2: 4 buttons to select the size (no invalid input possible)

REQ-3: Must create a drawing area with the designated size

REQ-4: Any frames added will be the same size

## Add/Delete Frames

4.2.1 Description and Priority

Adjusts the number of frames for the sprite animation. This will be incremental, for example "add a frame" and "delete a frame". The “add a frame” will only add a frame to the end of the frame sequence, i.e. a user cannot change the order of the frames. The user can delete any frame, no matter the order.

Priority: High

4.2.2 Stimulus/Response Sequences

A bar along the bottom will show each of the existing frames, have a button for adding a new frame, and have buttons below each of the frames to delete that frame.

4.2.3 Functional Requirements

REQ-1: Bar to show a thumbnail of each frame with scrolling capabilities

REQ-2: Clicking the add frame button will add a new frame to the bar and change the frame in the drawing window to the newly made frame

REQ-3: Clicking the delete button below an individual frame will remove that frame from the bar (if active frame is the deleted frame, new active frame is the previous frame)

## Animation Preview

4.3.1 Description and Priority

Allows the user to view the different frames in a cycle (and thus the animation).

Priority: Medium

4.3.2 Stimulus/Response Sequences

The animation will continuously play. The speed at which it plays will be determined by the frames per second; there will be a slider that the user can drag to increase and/or decrease the frames per second. The cycle will happen automatically.

4.3.3 Functional Requirements

REQ-1: The preview will happen automatically as the user creates additional frames.

REQ-2: Bad inputs for the frames per second will not be allowed, as the slider will limit the inputs to an acceptable range.

## Pen

4.4.1 Description and Priority

A simple pen tool for drawing sprites. This is the most fundamental drawing tool.

Priority: High

4.4.2 Stimulus/Response Sequences

The user will click and drag the mouse to draw, after selecting a color.

4.4.3 Functional Requirements

REQ-1: Changing the color of the pixels around where the mouse is clicking

REQ-2: Color picker for the user to choose a color (see item 4.7)

## Shape Tool

4.5.1 Description and Priority

Allows the user to draw a square, circle, or straight line.

Priority: Low

4.5.2 Stimulus/Response Sequences

The user will click and drag the mouse to draw their selected shape. The size will be determined by how far the mouse varies from the location of when the mouse was clicked.

4.5.3 Functional Requirements

REQ-1: Preview line is displayed while user drags mouse across the drawing window

REQ-2: User unclicks, and their shape is presented on the drawing window

REQ-3: If the shape does not match the user’s expectation, user will need to use the eraser tool to adjust or discard their shape drawing

## Eraser

4.6.1 Description and Priority

Changes selected pixels back to blank.

Priority: Medium

4.6.2 Stimulus/Response Sequences

Selecting the eraser tool and then clicking and dragging to select the pixels.

4.6.3 Functional Requirements

REQ-1: Button to select eraser tool

## Color Picker

4.7.1 Description and Priority

An image display of a color gradient with a selector that the user can drag around. This is necessary for users to make quality images.

Priority: High

4.7.2 Stimulus/Response Sequences

Within the color picker, the user drags the circle around the image to select the color at that point. This color will automatically apply to the pen, shape, and fill-bucket tools.

4.7.3 Functional Requirements

REQ-1: Color gradient image

REQ-2: Draggable selector

## Fill Bucket

4.8.1 Description and Priority

Allows the user to fill all adjacent pixels of the same color with a desired color.

Priority: Medium

4.8.2 Stimulus/Response Sequences

Selecting the fill bucket tool and clicking the desired area to fill will change the pixel clicked as well as any adjacent pixels of the same color to the selected color in the color picker.

4.8.3 Functional Requirements

REQ-1: Button to select the fill bucket tool

REQ-2: Color picker for the user to choose a color (see item 4.7)

## Save and load: *example of Json on next page*

4.9.1 Description and Priority

Allows the user to open an existing sprite to edit and save a sprite they have made. This, particularly the save option, is necessary for someone to make use of the things they create.

Priority: High

4.9.2 Stimulus/Response Sequences

File dropdown menu to load an existing file and to save one that has been created.

4.9.3 Functional Requirements

REQ-1: Access to the file dropdown menu

REQ-2: Parsing an existing file to upload frames to the editor

REQ-3: Save sprite with file extension as .ssp

Json Example  
  
{

"file\_name": "fileNameExample",

"sprite\_width”: 2,

“sprite\_height”:2,  
 “frame\_count”: 2,

"file\_frames": [  
 "img1": [

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

}

],

"img2": [

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

},

{

"red": 255,

"green": 0,

"blue": 0,

"alpha": 255

}

]  
 ]

}

}