

Proyecto II – Paint++

Instituto Tecnológico de Costa Rica
Área Académica Ingeniería en Computadores
CE2103 - Algoritmos y Estructuras de Datos II
Primer Semestre 2022
Valor: 20%



Objetivo General

- Desarrollar una aplicación utilizando el lenguaje de programación C++.

Objetivos Específicos

- Aplicar conceptos de pathfinding y manipulación de matrices.
- Investigar y desarrollar una aplicación en el lenguaje de programación C++
- Investigar sobre manipulación de imágenes en C++
- Aplicar patrones de diseño en la solución de un problema.

Descripción del Problema

Se le solicita implementar un editor de imágenes (en formato Bitmap/BMP) similar a Paint. En términos generales, Paint++ debe ser capaz de permitir al usuario dibujar sobre un canvas con ciertas funcionalidades y aplicar manipulaciones sobre dicho canvas. Igualmente permite cargar imágenes en formato BMP y manipularlas. A continuación se listan las características:

- El usuario puede crear un canvas de un tamaño en pixeles definido por el mismo usuario.
- El sistema permite agregar figuras geométricas predefinidas. El usuario puede arrastrar dichas figuras y cambiar el tamaño
- El sistema es capaz de borrar trazos o figuras completas utilizando la función de borrador. El tamaño del borrador puede cambiarse a criterio del usuario
- El sistema permite dibujar utilizando la función de lápiz. El usuario puede seleccionar el color y el ancho de los trazos.
- El sistema permite dibujar líneas rectas utilizando la función de lapicero. El usuario selecciona dos puntos en el canvas y se dibuja la línea. Al igual que la función de lápiz, usuario puede seleccionar el grosor y color de los trazos
- El sistema provee la función de color picker para determinar el color RGB de un pixel seleccionado
- El sistema provee la función paint-fill para pintar una región. El usuario da click en algún píxel de la imagen y aplicando BFS, se pintan los pixeles alrededor hasta que el área circundante sea de otro color.
- El sistema provee la opción de rotar el canvas/imagen al igual que la opción de hacer flip horizontal y vertical
- El sistema lleva el registro de todos los comandos aplicados, por lo que usuario puede hacer undo y redo.
- El sistema debe proveer una funcionalidad de zoom.
- El sistema provee una funcionalidad para seleccionar secciones de la imagen.
 - Selección rectangular
 - Selección free-form
 - Selección mágica (similar a paint fill pero para seleccionar)
- Debe ser posible salvar el canvas en un archivo BMP y volver a abrirlo posteriormente así como cargar

imagenes BMP construidas fuera de Paint++

- Debe ser posible **aplicar un filtro de escala de grises a la imagen y un filtro negativo**
- **Puntos extra:** diseñar e implementar dos filtros adicionales para las imágenes.
- **No se permite el uso de OpenCV o cualquier otra biblioteca de manipulación de imágenes.**

Documentación requerida

1. Internamente, el código se debe documentar siguiendo los estándares de documentación para C++.
2. La documentación externa se hará en un documento que incluya lo siguiente (deberá entregarse un PDF):
 - a. Breve descripción del problema.
 - b. Diagrama de clases.
 - c. Descripción de las estructuras de datos desarrolladas.
 - d. Descripción de los algoritmos implementados para cada funcionalidad
 - e. Problemas encontrados en forma de bugs de *github*: En esta sección se detalla cualquier problema que no se ha podido solucionar en el trabajo.
3. **Planificación y administración del proyecto:** se utilizará Jira para la administración del proyecto. Debe incluir:
 - a. Lista de features e historias de usuario identificados de la especificación.
 - b. Plan de iteraciones que agrupen cada bloque de historias de usuario de forma que se vea un desarrollo incremental
 - c. Descomposición de cada user story en tareas.

Aspectos operativos y evaluación:

1. **Fecha de entrega: De acuerdo con el cronograma del curso y lo establecido en el TEC Digital**
2. **Es obligatorio usar Linux para este proyecto**
3. El proyecto tiene un valor de 20% de la nota del curso.
4. El trabajo es en **parejas**.
5. Deben entregar en el TEC Digital/classroom (según aplique) un documento con el link del repositorio de GitHub, Jira y el PDF de la documentación. En ambas herramientas deben dar acceso al correo del profesor.
6. Es obligatorio utilizar un Git y GitHub para el control de versiones del código fuente y evidenciar el uso de Commits frecuentes.
7. Es obligatorio integrar toda la solución.
8. El código tendrá un valor total de 70%, la documentación externa 20% y la defensa un 10%.
9. De las notas mencionadas en el punto anterior se calculará la Nota Final del Proyecto.
10. Se evaluará que la documentación sea coherente, acorde a la dificultad/tamaño del proyecto y el trabajo realizado. Se recomienda que realicen la documentación conforme se implementa el código.
11. La nota de la documentación externa es proporcional a la completitud del proyecto.
12. Las citas de revisión oficiales serán determinadas por el profesor durante las lecciones o mediante algún medio electrónico.
13. Los estudiantes pueden seguir trabajando en el código hasta 15 minutos antes de la primera cita de revisión oficial.
14. Aún cuando el código y la documentación externa tienen sus notas por separado, se aplican las siguientes restricciones
 - a. Si no se entrega documentación externa, automáticamente se obtiene una nota de cero en la nota final del proyecto.

- b. **Si no se entrega el Anexo se obtiene una nota de cero en la nota final del proyecto. El Anexo debe entregarse completo. Si tienen alguna duda con respecto a las partes del documento deben aclararlo con el profesor.**
 - c. Si no se utiliza un manejador de código se obtiene una nota de cero en la nota final del proyecto.
 - d. Si la documentación externa no se entrega en la fecha indicada se obtiene una nota de cero en la nota final del proyecto.
 - e. Si el código no compila se obtendrá una nota de cero en la nota final del proyecto, por lo cual se recomienda realizar la defensa con un código funcional.
 - f. El código debe ser desarrollado en C++ (Linux), en caso contrario se obtendrá una nota de cero en la nota final del proyecto.
15. La revisión de la documentación será realizada por parte del profesor. Podría ser revisada, por parte del profesor, antes o después de la cita de revisión del proyecto. Durante la defensa del proyecto sí se revisará el diagrama de clases, la documentación interna y la documentación en el manejador de código.
16. Cada estudiante tendrá como máximo 15 minutos para exponer su trabajo al profesor y realizar la defensa de éste, es responsabilidad de los estudiantes mostrar todo el trabajo realizado, por lo cual se recomienda tener todo listo antes de ingresar a la defensa.
17. Cada excepción o error que salga durante la ejecución del proyecto y que se considere debió haber sido contemplada durante el desarrollo del proyecto, se castigará con 2 puntos de la nota final del proyecto.
18. Cada estudiante es responsable de llevar los equipos requeridos para la revisión, si no cuentan con estos deberán avisar al menos 2 días antes de la revisión al profesor para coordinar el préstamo de estos.
19. Durante la revisión únicamente podrán participar el estudiante, asistentes, otros profesores y el coordinador del área.

ANEXO DEL PROYECTO

Objetivo General

- Elaborar un documento que evidencie la participación activa de los integrantes del equipo de acuerdo con el rol asignado y la ejecución adecuada de los lineamientos establecidos por el grupo para el trabajo en equipo.

Objetivos Específicos

- Participar activamente en el equipo de trabajo de acuerdo con el rol asignado.
- Ejecutar adecuadamente los lineamientos para el trabajo en equipo (metas, roles, reglas, cronogramas, bitácoras, otros).

Atributos de Acreditación

- Trabajo individual y en equipo (Medio).

Descripción del Entregable

Cada grupo debe elaborar un documento que tenga la siguiente estructura:

1. Portada.
2. Tabla de contenidos.
3. Introducción.

4. Trabajo individual y en equipo (evidencia de la participación activa en el equipo de trabajo de acuerdo con el rol asignado y la ejecución adecuada de los lineamientos para el trabajo en equipo (metas, roles, reglas, cronogramas, bitácoras, otros):
 - a. Metas del proyecto: Describir los entregables asociados al proyecto.
 - b. Roles: Describir los roles y responsabilidades utilizados en el desarrollo del proyecto.
 - c. Reglas: Describir las reglas principales que han sido definidas para el trabajo en equipo.
 - d. Cronograma: Plan de proyecto con las actividades planeadas, responsables de cada actividad y fechas de entregas estimadas.
 - e. Minutas de las sesiones de trabajo: De manera que se evidencie la participación activa en el equipo de trabajo de acuerdo con el rol asignado y el seguimiento al plan de trabajo.
 - f. Bitácora: Evidencia la participación activa en el trabajo en equipo y la ejecución adecuada de los lineamientos para el trabajo en equipo, de cada uno de los integrantes del equipo. Deben describir las actividades realizadas como reuniones con el compañero de trabajo, investigaciones, consultas, entre otras. Se debe describir todo por más insignificante que sea. Esto demostrará el trabajo de cada uno de los miembros del equipo según el rol asignado.

Aspectos operativos y evaluación

1. El documento debe ser enviado en formato PDF en conjunto con los entregables del proyecto.