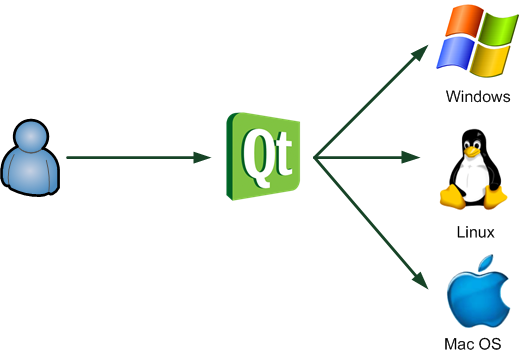
# KẾT QUẢ THỰC NGHIỆM

## Môi trường kiểm tra thuật toán

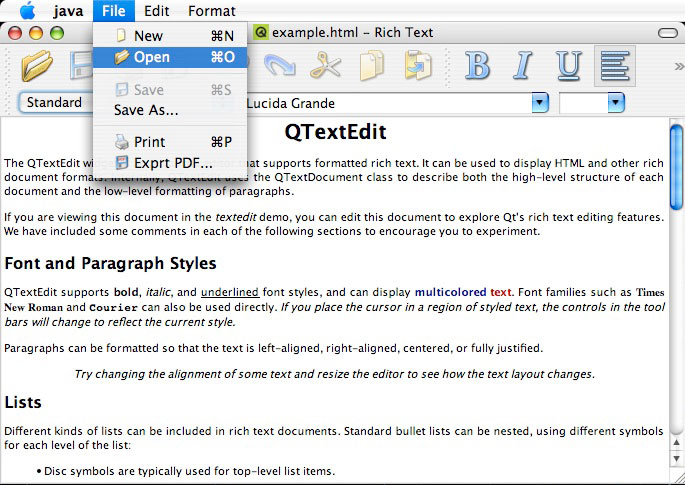
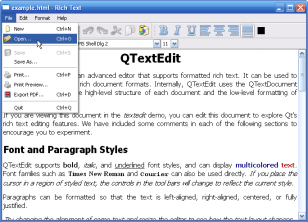
**Qt là một framework** đa nền tảng, được tích hợp trên **nhiều hệ điều** hành như:

* Windows
* Linux
* Mac Os



Hình 6‑1 Qt được tích hợp trên nhiều hệ điều hành

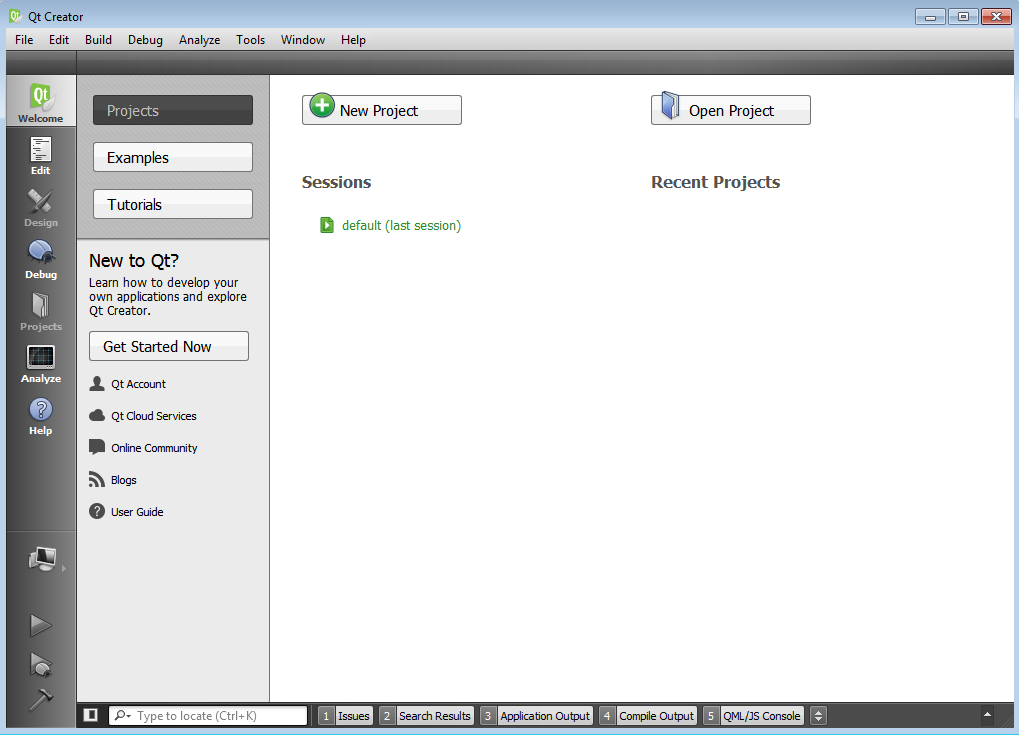
Khi người lập trình viên viết bằng Qt và Qt được dịch các câu lệnh ra tùy theo hệ điều hành. Dựa trên cơ chế này, cửa sổ mà người lập trình viên tạo ra sẽ thay đổi phù hợp với từng hệ điều hành như trong dưới đây:



Hình 6‑2 Cửa sổ thay đổi tùy theo hệ điều hành

Trong các dự án Qt người phát triển ứng dụng có thể sử dụng nhiều môi trường (IDE) để biên dịch và chạy thuật toán như: Qt Creator, Visual Studio for Qt5,…. Qua quá trình tìm hiểu thì chúng em nhận thấy **Qt Creator** tích hợp nhiều công cụ như:

* Chương trình viết mã lệnh (Code editor)
* Chương trình sửa lỗi (Debugger)
* Chương trình thiết kế giao diện bằng thao tác kéo thả



Hình 6‑3 Giao diện của Qt Creator

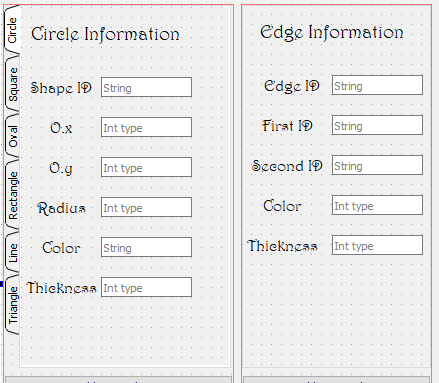
Qt Creator mang đến một giao diện rõ ràng, đơn giản, cho phép người dùng thiếp lập một số dự án mới bằng cách sử dụng phương pháp từng bước tiện dụng, đặc biệt đối với người chưa có kinh nghiệm vì họ sẽ được trợ giúp trong toàn bộ quá trình.

Ngôn ngữ lập trình với Qt Creator: **C++.**

## Bộ dữ liệu đầu vào

### Dữ liệu được nhập từ bàn phím

Trên giao diện của mình chúng em có tạo giao diện để người dùng có thể nhập trực tiếp dữ liệu, thông số của một hình bất kỳ từ bàn phím mà họ muốn sử dụng, thực hiện các chức năng của chương trình.



Hình 6‑4 Giao diện nhập dữ liệu từ bàn phím

Giao diện nhập dữ liệu từ bàn phím gồm ba phần chính:

* Chọn hình để nhập các thông số dữ liệu của hình
* Nhập thông số dữ liệu cho các hình
* Nhập thông số dữ liệu cho các cạnh nối tâm giữa hai hình

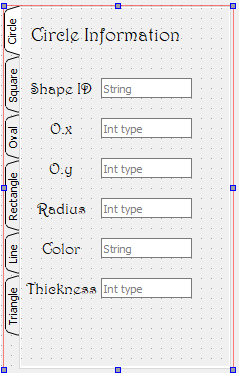
***Chọn hình để nhập dữ liệu***

Cho phép người dùng có thể click vào các tab: Circle, Square, Oval, Rectangle, Line, Triangle để có thể nhập dữ liệu hình tương thích.

***Thông số dữ liệu đầu vào cho shape***

Trong giao diện này các hình: Circle, Square, Oval, Rectangle, Line, Triangle chúng có các thuộc tính chung đó là:

* Tên hình: Shape ID
* Tọa độ tâm: O.x, O.y
* Màu sắc: Color
* Độ dày: Thickness



Hình 6‑5 Nhập dữ liệu cho Shape

Những thuộc tính chung của các hình được thể hiện trong bảng dưới đây:

|  |  |  |
| --- | --- | --- |
| Đối tượng | Thuộc tính chung | Kiểu dữ liệu |
| Shape | Shape ID | String |
| O.x | Int |
| O.y | Int |
| Color | String |
| Thickness | Int |

Bảng 1 Thuộc tính chung của Shape

Những thuộc tính cụ thể của từng hình: Circle, Square, Rectangle, Oval, Line, Triangle được thể hiện trong bảng dưới đây:

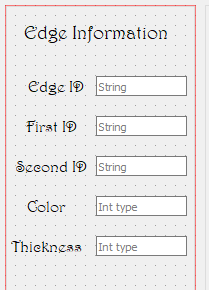
|  |  |  |
| --- | --- | --- |
| Đối tượng | Thuộc tính riêng | Kiểu dữ liệu |
| Cirle | Radius | Int |
| Square | Edge | Int |
| Oval | Width | Int |
| Heigth | Int |
| Rectangle | Width | Int |
| Heigth | Int |
| Line | A.x, A.y | Int |
| B.x, B.y | Int |
| Triangle | A.x, A.y | Int |
| B.x, B.y | Int |
| C.x, C.y | Int |

Bảng 2 Thuộc tính riêng của từng Shape

Khi người dùng nhập dữ liệu, mỗi hình sẽ có từng thuộc tính riêng biệt của nó nhưng vẫn có các thuộc tính chung của hình. Vd: Hình tròn (Circle) sẽ có thuộc tính riêng là Radius nhưng Hình vuông (Square) sẽ không có thuộc tính đó mà nó có thuộc tính Edge. Thuộc tính riêng của từng hình sẽ quyết định hình đó sẽ là hình gì, như thế nào.

***Thông số dữ liệu đầu vào cho edge***

Người dùng còn có thể nhập dữ liệu đầu vào để có thể tạo cạnh nối tâm giữa hai hình



Hình 6‑6 Nhập dữ liệu đầu vào của Edge

Các thuộc tính của cạnh nối tâm được thể hiện trong bảng dưới đây:

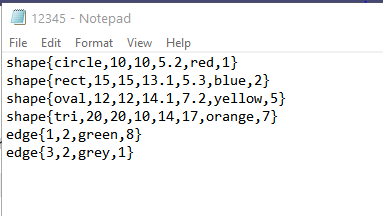
|  |  |  |
| --- | --- | --- |
| Đối tượng | Thuộc tính | Kiểu dữ liệu |
| Edge | Edge ID | String |
| First ID | Sting |
| Second ID | String |
| Color | Int |
| Thickness | Int |

Bảng 3 Thuộc tính của Edge

Cạnh nối tâm chỉ tạo khi người dùng đã tạo hai hình trước đó.

### Dữ liệu được nhập từ file text

Bộ dữ liệu đầu vào có thể được truyền vào từ file text dạng .txt



Hình ‑ Bộ dữ liệu từ file text

Một file text có định dạng như sau:

* Mỗi dòng chứa các thông số của Shape hoặc Edge.
* Các thông số được đặt trong dấu ngoặc nhọn {}.
* Các thông số cách nhau bởi dấu phẩy (,).
* Thông số đầu tiên là loại shape n.
* Tùy thuộc vào thông số đầu tiên về loại shape các thông số sau tương ứng với vị trí, kích thước, mầu và độ dầy cạnh.
* Dòng chứa cạnh nối shape có 2 thông số đầu tiên là index của shape (bắt đầu từ 1).

## Kết quả

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Tóm tắt kết quả đạt được và các vấn đề tồn tại

* Kết quả đạt được:
* Đã thiết kế được giao diện thực hiện đúng và đầy đủ các tính năng mà thầy yêu cầu.
* Các thành viên trong nhóm đã giúp đỡ nhau, đóng góp ý kiến để hoàn thành nội dung đề tài.
* Qua môn đồ án I được sự dẫn dắt của thầy Nguyễn Đức Minh, mọi người đã hiểu sâu hơn, đầy đủ hơn về ngôn ngữ lập trình C++ và OOP, biết sử dụng các thuật toán trong C++. Từ đó vận dụng vào thiết kế giao diện bằng framwork Qt và đặc biệt sử dụng mô hình M-V-C trong quá trình thiết kế giao diện.
* Các vấn đề tồn tại:
* Sản phẩm cần được thiết kế đẹp hơn, ngọn ngàng hơn, để người dùng có thể dễ dàng sử dụng các chức năng.
* Chưa tối ưu hóa code.
* Kỹ năng phân tích, giải quyết vấn đề còn thiếu.
* Kỹ năng làm việc nhóm cần được cải thiện.

## Bài học kinh nghiệm

* Qua đây chúng em rút ra được một số kinh nghiệm:
* Viết code chỉ cần sai lỗi nhỏ, thiếu một dấu phẩy thì nó cũng không chạy.
* Cần command code để mọi người trong team cũng như người xem có thể đọc hiểu được.
* Cần cải thiện một số kỹ năng như: làm việc nhóm hiệu quả, kỹ năng tìm kiếm tài liệu, cần học tiếng anh thật tốt.

## Hướng phát triển

* Mở rộng thêm các chức năng của giao diện để người dùng có thêm sự lựa chọn.
* Thiết kế giao diện đăng nhập để tăng tính bảo mật cho ứng dụng.