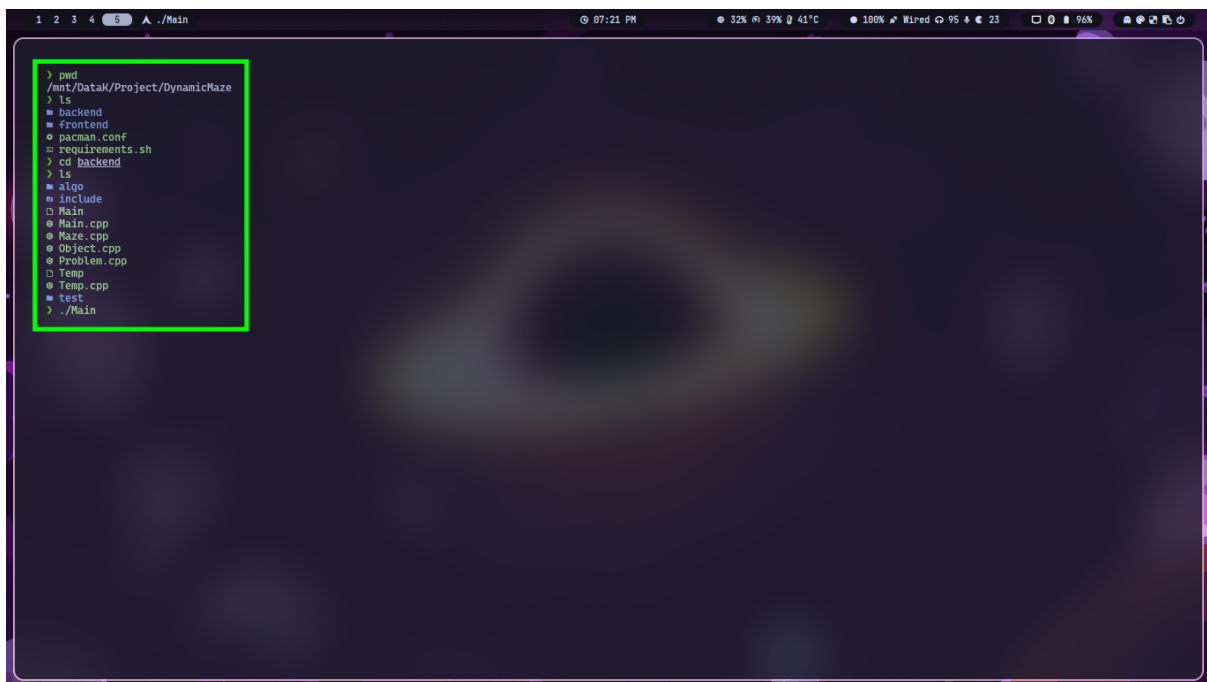


Hướng dẫn sử dụng DynamicMaze App

Sau khi đã hoàn thành thành công việc cài đặt WSL cho Arch Linux (nếu bạn chỉ có thể dùng trên Windows), NodeJS cho Frontend, C++ và các thư viện C++ liên quan cho Backend thì chúc mừng các bạn đã có thể sử dụng sản phẩm của chúng tôi.

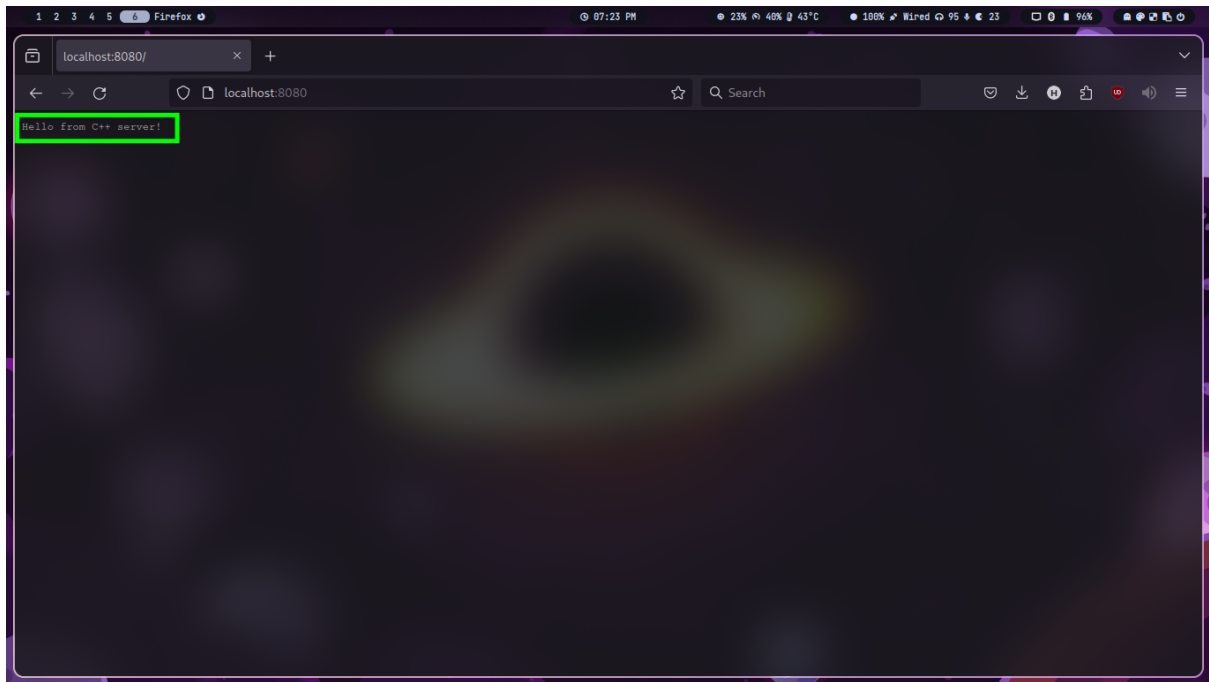
Bước đầu tiên để bắt đầu dịch vụ backend, các bạn:

- + Di chuyển đến folder đã git clone dự án của chúng tôi. Mặc định theo hướng dẫn chúng tôi clone trên máy các bạn (và các bạn nên để như vậy) tại vị trí ~/. Vị trí này là quan trọng vì script cài đặt có lấy nhiều thông tin để tự động cài các package quan trọng cho các bạn rồi.
- + Sau đó các bạn cd vào thư mục backend. Rồi cấp quyền thực thi và chạy file start Backend bằng lệnh `./Main`.
- + Mọi người có thể xem hình ảnh ví dụ sau để xác định có khởi động dịch vụ backend được chưa.



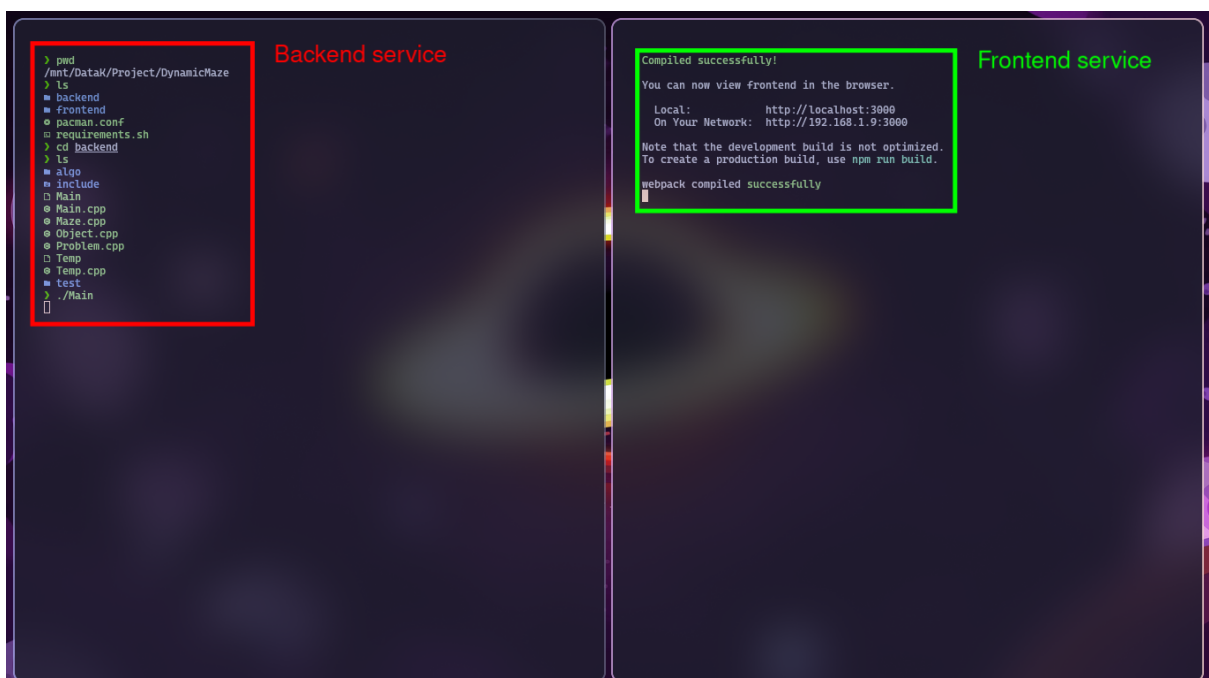
```
> pwd
/mnt/DataK/Project/DynamicMaze
> ls
backend
frontend
pacman.conf
requirements.sh
> cd backend
> ls
algo
include
Main
Main.cpp
Maze.cpp
Object.cpp
Problem.cpp
Temp
Temp.cpp
test
> ./Main
```

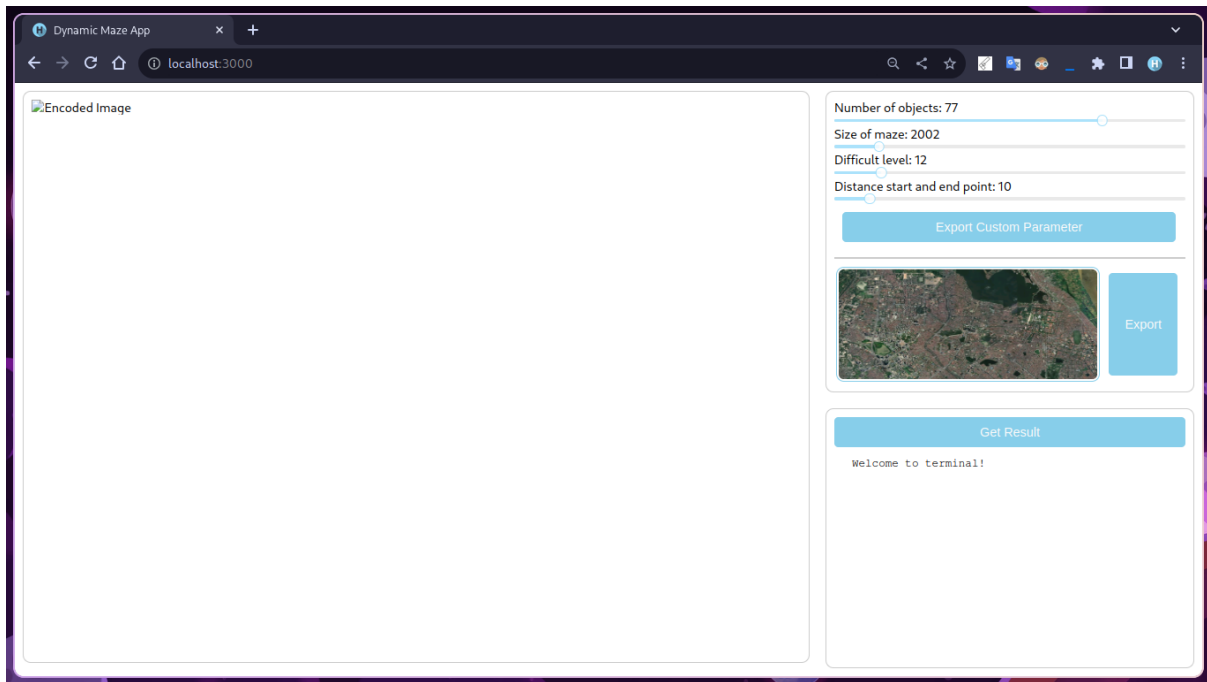
- + Mọi người có thể vào trình duyệt gõ localhost:8080/ để xem trạng thái của backend. Địa chỉ <http://localhost:8080/> đã được chúng tôi thiết lập 1 API để khi các bạn truy cập sẽ hiển thị trạng thái server được viết bằng C++. Thông báo **Hello from C++ server!** hiện ra có nghĩa là backend đã khởi động thành công và không có lỗi phát sinh.



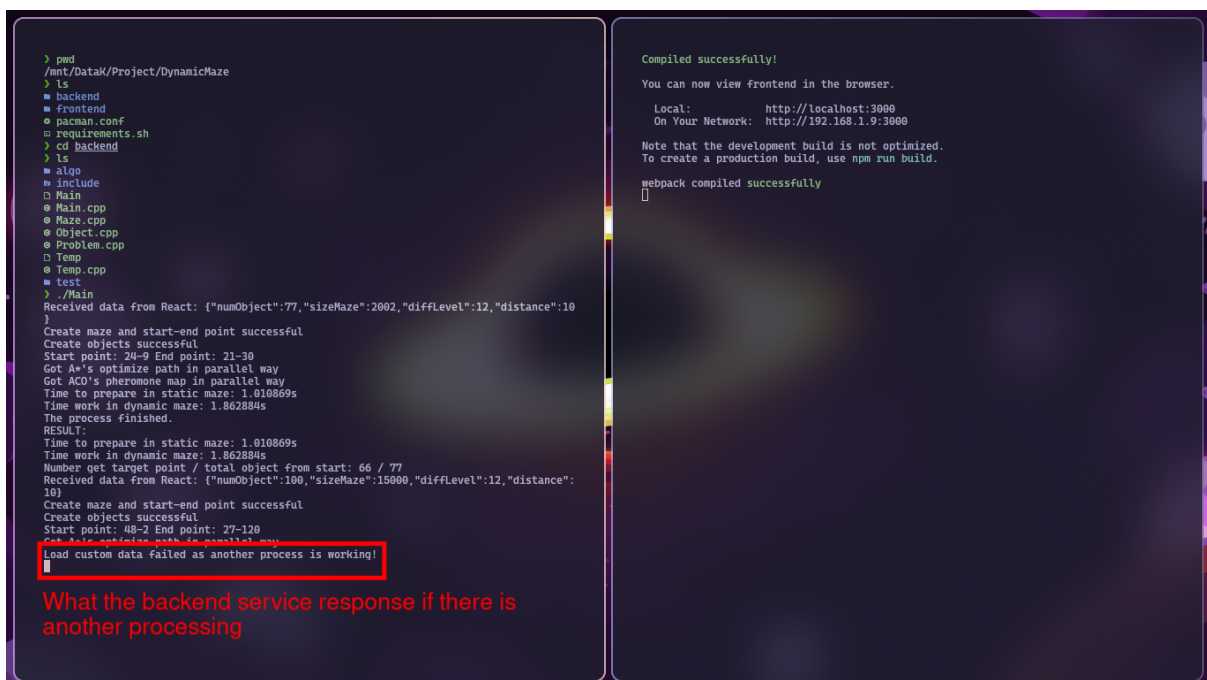
- + Bây giờ mọi người không được đóng terminal chứa tiến trình backend đang chạy. Bởi vậy mọi người cần tạo terminal mới cũng đã đến đường dẫn của app chúng tôi. Bây giờ cd vào thư mục frontend. Nếu mọi người đã thực hiện npm i để cài đặt các thư viện liên quan của React (do chuẩn thiết kế pull code của github - chủ yếu không muốn người dùng download thư mục nặng nề này về nên mục này được không được chúng tôi tải lên github), vậy thì để start dịch vụ frontend các bạn chỉ cần đơn giản gõ: `npm start`.

Kết quả là dịch vụ frontend được bắt đầu ở cổng 3000. Tức truy cập vào <http://localhost:3000> và mọi người sẽ thu được trang web frontend.

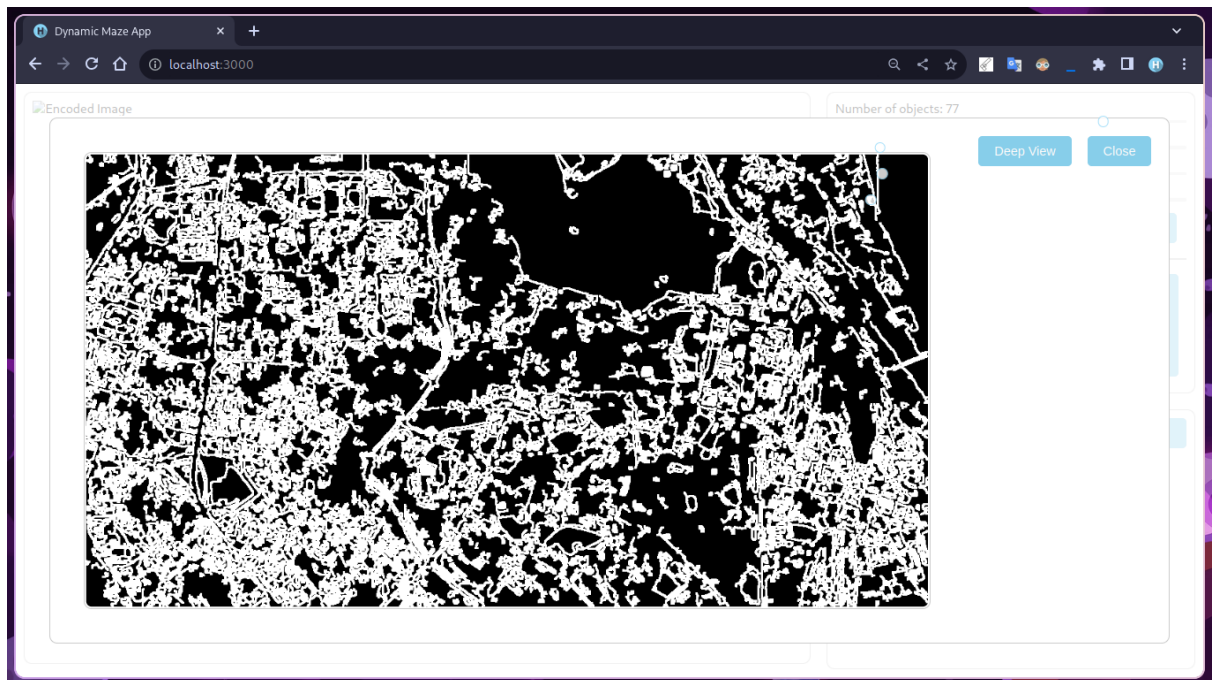




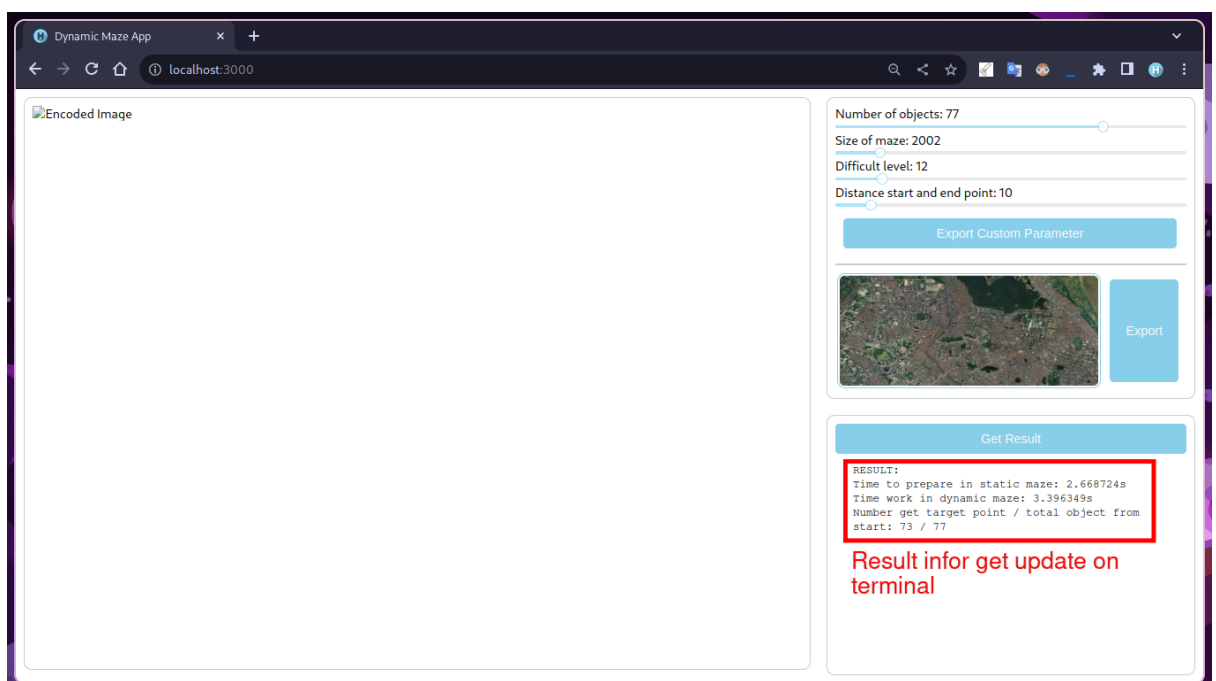
- Đây là hình ảnh trang web của chúng tôi. Từ trái qua phải, từ trên xuống dưới chúng tôi có bảng hiển thị các frame ảnh mô tả cấu trúc mê cung thay đổi và sự di chuyển của các object. Tiếp theo là trung tâm điều khiển với các tham số tùy chỉnh của bài toán, sau khi bấm button thì data sẽ truyền từ frontend về cho backend thông qua các API. Nếu các bạn bấm nhiều lần button thì sẽ có kết quả thông báo ở backend là có tiến trình khác đang thực hiện. Đây là cách chúng tôi xử lý ngoại lệ tránh trường hợp máy tính người dùng quá yếu và vì sự cố gì đó người dùng liên tục send request giải bài toán (vốn giao diện người dùng được thiết kế để hiển thị tiến trình giải và kết quả của 1 request).



- Ngoài ra ở trung tâm điều khiển các bạn có thể thử qua chế độ try hard của chúng tôi khi chúng tôi đã import và biến bản đồ Hanoi thành 1 mê cung khổng lồ phục vụ ứng dụng bài toán.



- Sau khi hoàn thiện lời giải, mọi người có thể bấm vào Get result để cập nhật kết quả thống kê lên terminal.



Linux rất ảo diệu khi chạy các dịch vụ backend C++ và frontend JavaScript ngay trên cùng 1 máy tính có phải không. Chúc mọi người cài đặt và sử dụng thành công!