**BÁO CÁO ĐỒ ÁN LẬP TRÌNH GAME**



# Giới thiệu sơ lượt về ý tưởng game

# Kiến trúc tổng quan của game

# Chi tiết thiết kế các thành phần trong game

## Thiết kế màn hình menu chính

### Mô tả

Đây là màn hình hiện ngay khi vào game, chức năng hiện danh sách các menu chính cho người chơi chọn

## Thiết kế màn hình bản đồ

### Mô tả

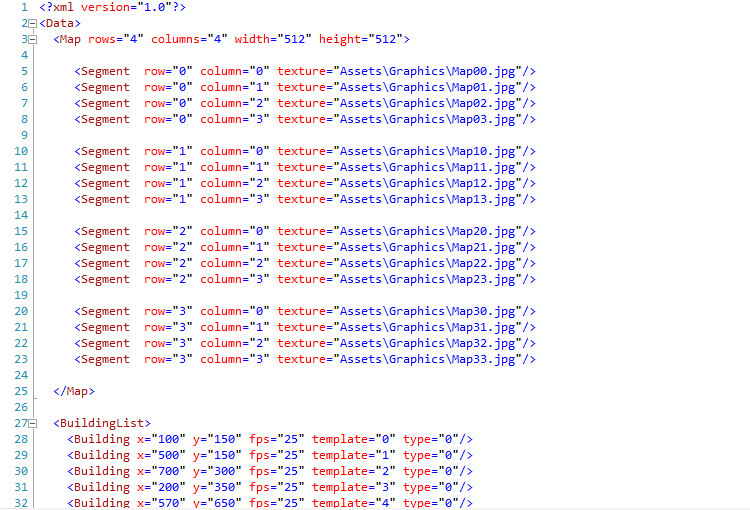
Chức năng chính của màn hình này là hiện các ngôi nhà, pháo đài cho người chơi đi đến. Tại mỗi pháo đài sẽ có chức năng khác nhau như nhận một nhiệm vụ, chiến đấu với một màn mini-game hoặc xem thông tin kỹ năng, trang bị.



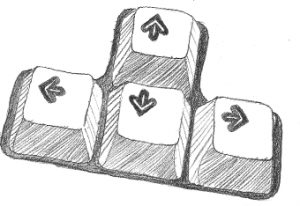
### Chi tiết các chức năng

#### Chức năng load map từ file xml:

Thông tin về map được lưu trong file xml gồm có các mảnh kiến tạo map, các đối tượng được hiện trên map như công trình…



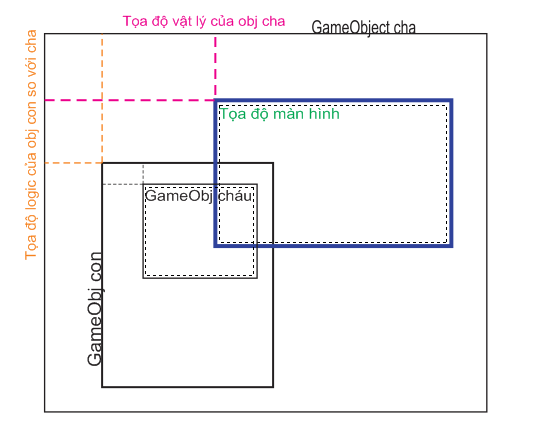
#### Hệ thống cuộn map

Bằng cách sử dụng các phím mũi tên để cuộn map lên, xuống, trái, phải để xem các vùng bản đồ. 

#### Hệ thống tọa độ logic và tọa độ thực

Các vật thể trong game có cả 2 hệ tọa độ có quan hệ với nhau nhầm đơn giải hóa quá trình lập trình di chuyển, cũng như có mức tiếp cận gần với thực tế hơn.

Tọa độ thực là tọa độ của vật thể đối với màn hình. Tức là tọa độ sẽ dùng để vẽ. Tọa độ logic là tọa độ của đối tượng so với đối tượng chứa nó. Do đó khi lập trình chỉ cần quan tâm đến tọa độ logic. Tọa độ thực sẽ được tính toán khi vẽ.

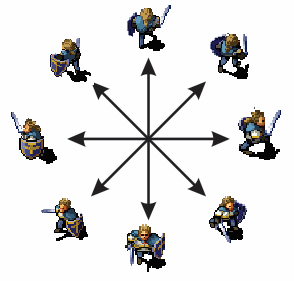


Với các GameObj root – có thuộc lớp GameObj nào cả sẽ được qui ước có tọa độ logic là tọa độ thực với màn hình.

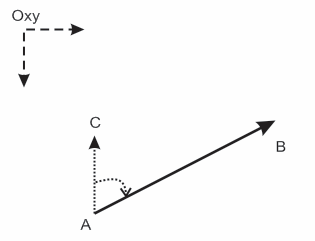
Các GameObj chỉ cần lưu giữ tọa độ Logic của nó, khi cần lấy tọa độ thực (để vẽ) có thể được xác định dễ dàng bằng tọa độ thực của đối tượng chứa nó và tọa độ logic của nó.

#### Hệ thống di chuyển có hướng của các Sprite

Các Sprite ngoài hiệu ứng di chuyển còn có di chuyển theo hướng. Tức mõi hướng sẽ có một hiệu ứng di chuyển khác nhau. Số hướng hỗ trợ sẽ phụ thuộc vào sprite.

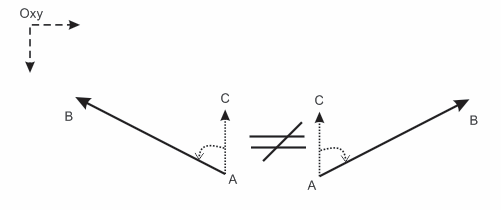


Với hướng di chuyển từ A đến B để xác định sử dụng sprite hướng nào thì trải qua quá trình tính toán sau:

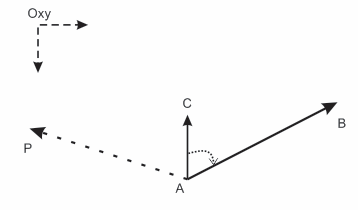


Với và góc tạo bởi 2 vector là góc α với

Nhưng có vấn đề là phân biệt giữa góc do định nghĩa góc giữa 2 vector. Nên hai hình dưới sẽ có cùng một góc α dẫn đến có cùng một hướng sprite nhưng phải là một hướng khác mới đúng.



Để giải quyết vấn đề cần xác định một điểm nằm bên trái hay bên phải một vector. Ở đây em dùng một cách mà trong khoa học robot vẫn dùng đó là tính tích hữu hướng của 2 vector trên:



Với giả thiết thì ta được

Khi đó nếu thì góc di chuyển so với trục 0h của sprite là ngược lại thì góc đó là .

## Thiết kế màn hình mini-game