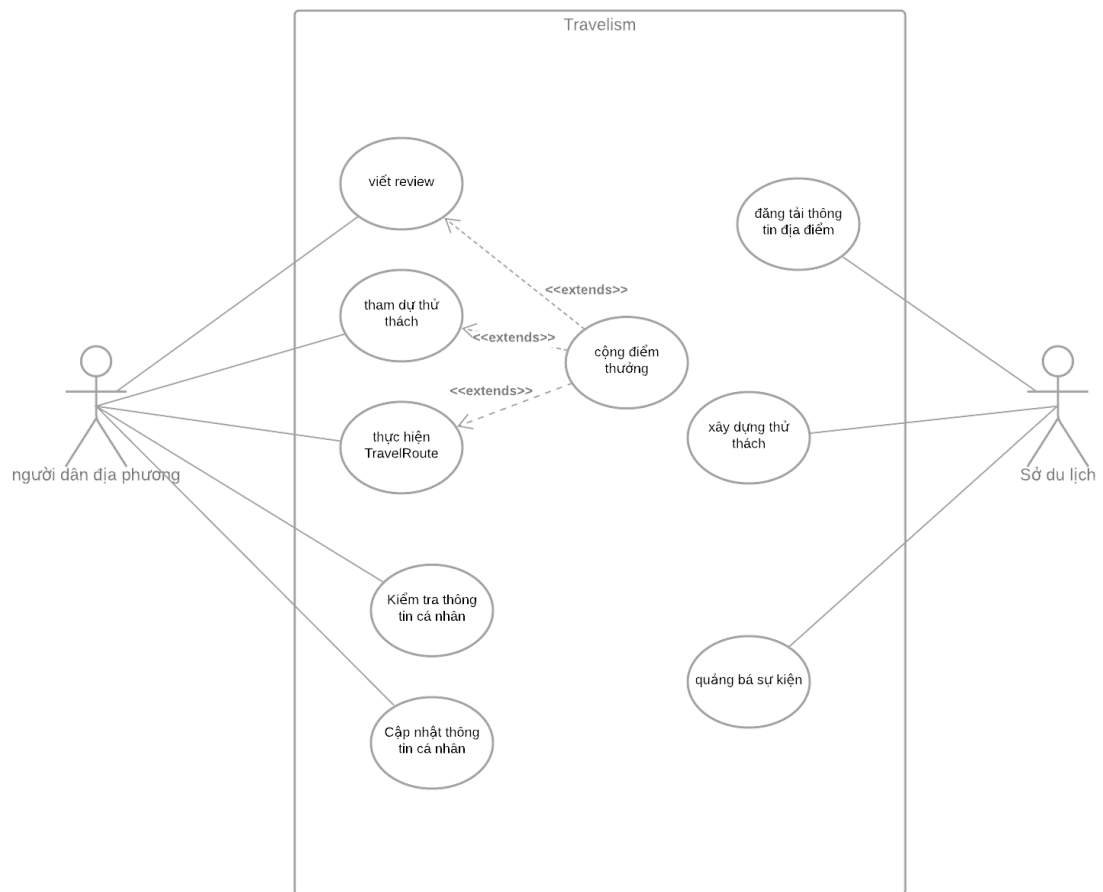


# Functional & Technical Specification

## Chức năng



Sơ đồ use case

Dựa vào sơ đồ use case ta có thể thấy nền tảng du lịch điểm thưởng Travelism sẽ có hai luồng người dùng. Đầu tiên là người đi du lịch cụ thể là những người địa phương đang sinh sống và làm việc nơi đó. Người dùng này sẽ có thể thực hiện các chức năng như viết review địa điểm, tham dự thử thách và thực hiện một TravelRoute. Các hoạt động này sẽ đem lại cho người dùng một số điểm thưởng loyalty point nhất định. Nhờ có những

điểm thưởng này mà người dùng du lịch có thể đổi các vật phẩm tương ứng.

Đối với loại người dùng là sở du lịch của các địa phương, họ sẽ có quyền thực hiện các chức năng như là đăng tải thông tin địa điểm, tạo các sự kiện và quảng bá nó và xây dựng các thử thách xung quanh địa điểm đó. Ngoài ra, sở du lịch sẽ kết hợp với bên làm dịch vụ để cùng nhau đưa các sản phẩm được bày bán tại các địa điểm lên ứng dụng để làm phần thưởng quy đổi điểm. Số lượng vật phẩm sẽ được giới hạn bởi sự thống nhất giữa sở du lịch cùng với bên dịch vụ. Việc đẩy các vật phẩm lên hệ thống blockchain để lưu trữ là một vấn đề do sẽ tốn rất nhiều phí transaction cho một lần lưu vật phẩm. Chính vì điều này, team Normie đề xuất việc lưu trữ thông tin vật phẩm ở trên các cơ sở dữ liệu thông thường. Tuy nhiên, làm sao để kết hợp được với smart contract để thực hiện các vấn đề liên quan tới điểm thưởng được lưu trên blockchain là một bài toán mà team Normie chưa tìm ra được giải pháp.

## **Công nghệ được dùng:**

Phiên bản web sẽ dùng ReactJS framework cho front-end. Với sự hỗ trợ của node package manager(npm) việc sử dụng các package khác như web3 trở nên thuận tiện và dễ dàng thiết lập hơn.

Solidity là ngôn ngữ chính để viết smart contract dành cho nền tảng. Smart contract sẽ có các chức năng cộng điểm cho người dùng khi họ viết bài review, hoàn thành thử thách cũng như hoàn thành một TravelRoute. Hệ thống tính điểm sẽ được lưu trữ trên blockchain đồng nghĩa với việc không ai có thể thay đổi nó. Smart contract sau đó sẽ được deploy bằng remix.ethereum trên mạng lưới injected web3. Với injected web3, ta có thể sử dụng package web3 để gọi và sử dụng smart contract đã được deploy. Để có thể thực hiện các chức năng ở smart contract, cần phải cấp cho

người dùng một tài khoản có chứa ethereum để thực hiện các chức năng vì mỗi chức năng đều tốn một lượng gas nhất định.

Với Metamask, ta sẽ tạo được một wallet. Tuy nhiên, wallet mà chúng ta sử dụng sẽ không có quá nhiều ETH để thực hiện các chức năng của smart contract. Vì vậy, ta sẽ dùng truffle để có thể tạo một wallet local với 10 tài khoản chứa 100 ETH mỗi cái.

-