

VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY  
HO CHI MINH UNIVERSITY OF TECHNOLOGY  
FACULTY OF COMPUTER SCIENCE & ENGINEERING



## ĐÁNH GIÁ HIỆU NĂNG HỆ THỐNG

---

**Assignment:**  
**THE QUEUING SYSTEMS-BASED  
PERFORMANCE EVALUATION PROJECT**

---

Lecturer: Trần Văn Hoài  
Class: L01 & L02  
Group: **Ghost Orchid**  
Member: Phan Thanh Huy - 2211257  
Phạm Việt Hùng - 2211349  
Hà Minh Nguyên - 2212288  
Nguyễn Trọng Đạt - 2210707  
Lý Vĩnh Khang - 1852443  
Lương Trung Tín - 1852203  
Đoàn Tường Chánh Đạo - 2210644  
Nguyễn Trường Thái Khang - 2211458

Thành phố Hồ Chí Minh, ngày 25 tháng 10 năm 2025

BÁO CÁO PHÂN CÔNG NHIỆM VỤ VÀ KẾT QUẢ THỰC HIỆN ĐỀ  
TÀI CỦA TỪNG THÀNH VIÊN

HỌ VÀ TÊN	MSSV	PHÂN CÔNG	MỨC ĐỘ ĐÓNG GÓP
Phan Thanh Huy	2211257	Platoon 1	100%
Phạm Việt Hùng	2211349	Platoon 3	100%
Lý Vĩnh Khang	1852443	Platoon 3	100%
Nguyễn Trọng Đạt	2210707	Platoon 1	100%
Đoàn Tường Chánh Đạo	2210644	Platoon 2	100%
Hà Minh Nguyên	2212288	Platoon 1	100%
Lương Trung Tín	1852203	Platoon 3	100%
Nguyễn Trường Thái Khang	2211458	Platoon 2	100%

Bảng 1: Phân công thành viên

Nhóm trưởng:

Thanh Huy

Phan Thanh Huy

# Mục lục

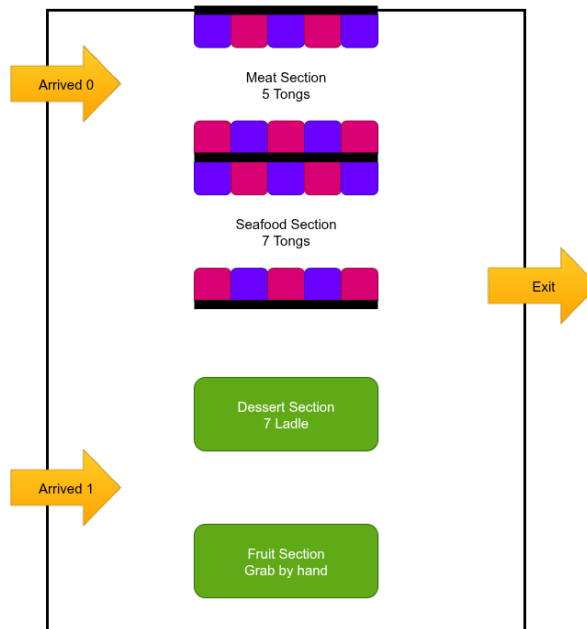
<b>1</b>	<b>Giới thiệu</b>	<b>1</b>
<b>2</b>	<b>Thiết kế hệ thống</b>	<b>1</b>
2.1	Tổng quan kiến trúc . . . . .	1
2.2	Các thành phần chính . . . . .	2
2.3	Triển khai các mô hình hàng đợi . . . . .	2
<b>3</b>	<b>Kiến trúc hệ thống</b>	<b>3</b>
3.1	Bản thiết kế mô hình hệ thống . . . . .	3
3.2	Mô tả các module và sự tương tác giữa chúng . . . . .	3
3.2.1	Lớp BuffetSystem (Hệ thống Nhà hàng) . . . . .	5
3.2.2	Lớp FoodStation (Quầy Thức ăn) . . . . .	5
3.2.3	Lớp Customer (Khách hàng) . . . . .	6
3.2.4	Lớp Analysis (Bộ Phân tích) . . . . .	6
3.2.5	Mối Quan Hệ (Relationships) . . . . .	6
<b>4</b>	<b>Các thành phần chính của mô hình</b>	<b>6</b>
<b>5</b>	<b>Luồng hoạt động cốt lõi</b>	<b>7</b>

## 1 Giới thiệu

Dự án thực hiện mô phỏng và đánh giá hiệu năng của các hàng đợi sử dụng thư viện SimPy, gồm các mục tiêu sau:

- Xây dựng các hàng đợi theo kí hiệu Kendall
- Mô phỏng hệ thống phục vụ tự chọn tại nhiều quầy
- So sánh kết quả mô phỏng với công thức toán học lý thuyết
- Đánh giá hiệu năng dưới các tải công việc khác nhau

**Trường hợp nghiên cứu: Nhà hàng buffet** Hệ thống mô phỏng tập trung vào một nhà hàng buffet, nơi khách hàng tự phục vụ tại nhiều quầy. Mô hình này sẽ giúp đánh giá hiệu năng phục vụ, thời gian chờ đợi của khách, và cách phân bổ quầy ảnh hưởng đến trải nghiệm khách hàng. Các quầy thức ăn được bố trí trong một căn phòng lấy thức ăn với hai lối vào và một lối ra, khách hàng chỉ có thể đi qua phòng này để lấy thức ăn một lần duy nhất.



Hình 1: Bản đồ sàn phòng lấy thức ăn buffet.

Các mô hình hàng đợi được triển khai:

- **Random Order Serving (ROS):** việc phục vụ được diễn ra hoàn toàn ngẫu nhiên.
- **Shortest Jobs First (SJF):** khách hàng nào muốn lấy ít thức ăn hơn sẽ được phục vụ sớm hơn. Tuy nhiên để tránh hiện tượng starvation (khách hàng muốn lấy nhiều thức ăn hơn phải chờ mãi) cần kiểm tra để phục vụ ngay những khách hàng đã chờ quá lâu.
- **Dynamic Server:** Thay vì phân phối cố định các server vào các hàng đợi thì tổng hợp các server và phục vụ chung có nhiều hàng đợi cùng lúc. Lúc này khách hàng nào đến trước sẽ được chọn phục vụ trước.

## 2 Thiết kế hệ thống

### 2.1 Tổng quan kiến trúc

Hệ thống được thiết kế theo mô hình hướng đối tượng với các thành phần chính:

- **BaseQueueSystem**: Lớp cơ sở trừu tượng cho tất cả các mô hình hàng đợi
- **QueueSystemFactory**: Factory pattern để tạo các hàng đợi khác nhau
- **TheoreticalCaculator**: Tính toán các giá trị lý thuyết
- **ValidationAnalyzer**: So sánh kết quả mô phỏng với lý thuyết
- **MultiQueueSystem**: Mô phỏng hệ thống đa hàng đợi

## 2.2 Các thành phần chính

### Arrival process

- Sinh khách hàng đến theo phân phối Poisson (exponential inter-arrival times)
- Tỷ lệ người đến:  $\lambda$

### Service process

- Thời gian phục vụ theo phân phối exponential
- Tỷ lệ phục vụ:  $\mu$
- Số lượng server:  $m$

### Queue management

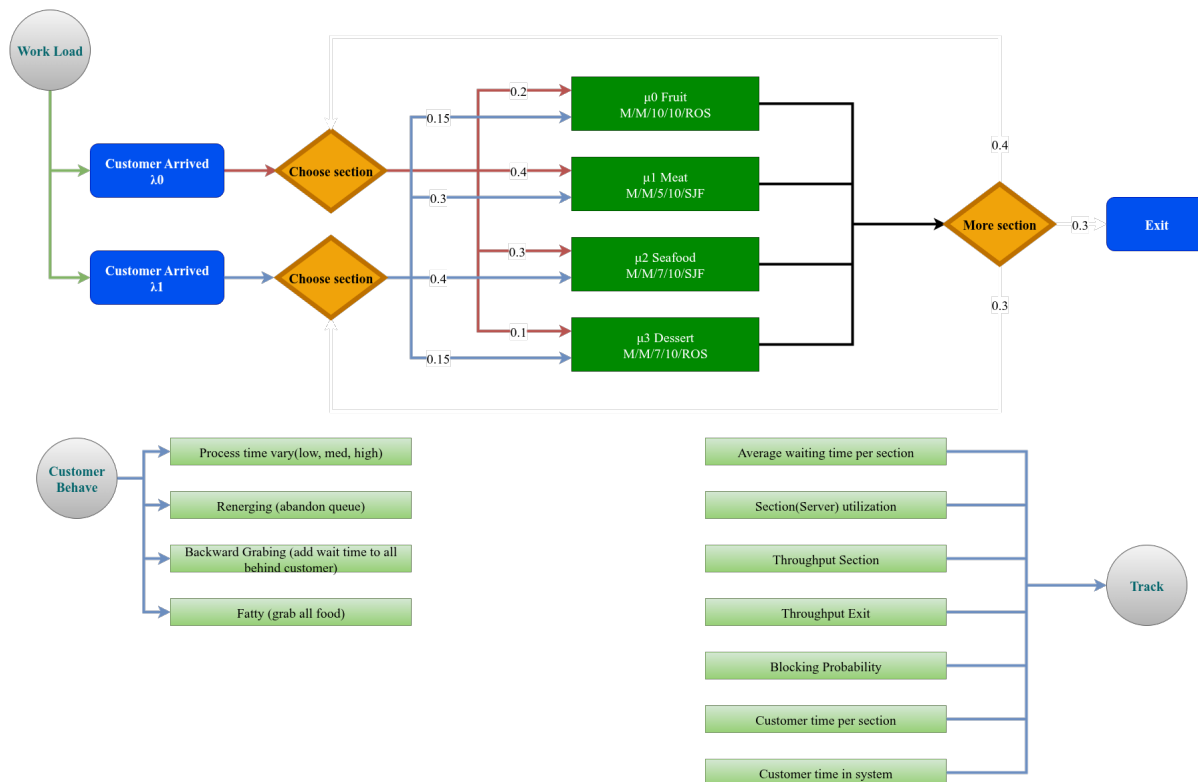
- Quản lý hàng đợi với dung lượng hữu hạn hoặc vô hạn
- Theo dõi thống kê: Thời gian chờ, thời gian phục vụ và độ dài hàng đợi

## 2.3 Triển khai các mô hình hàng đợi

- **ROS**: chọn khách hàng ngẫu nhiên trong hàng đợi.
- **SJF**: chọn khách hàng có thời gian phục vụ `serve_time` ngắn nhất để phục vụ trước, đồng thời kiểm tra nếu có khách hàng có thời gian đợi `wait_time` quá lâu phải đưa vào phục vụ ngay.
- **Dynamic server**: có một biến lưu trữ tổng số server, sau đó phân bổ cho các hàng đợi tùy vào số khách hàng. Một server sau khi phục vụ xong sẽ kiểm tra xem hàng đợi nào đông hơn (nhiều khách hàng hơn) để phục vụ cho khách hàng ở hàng đó. Chọn khách hàng để phục vụ trong hàng đợi theo nguyên tắc FIFO.

### 3 Kiến trúc hệ thống

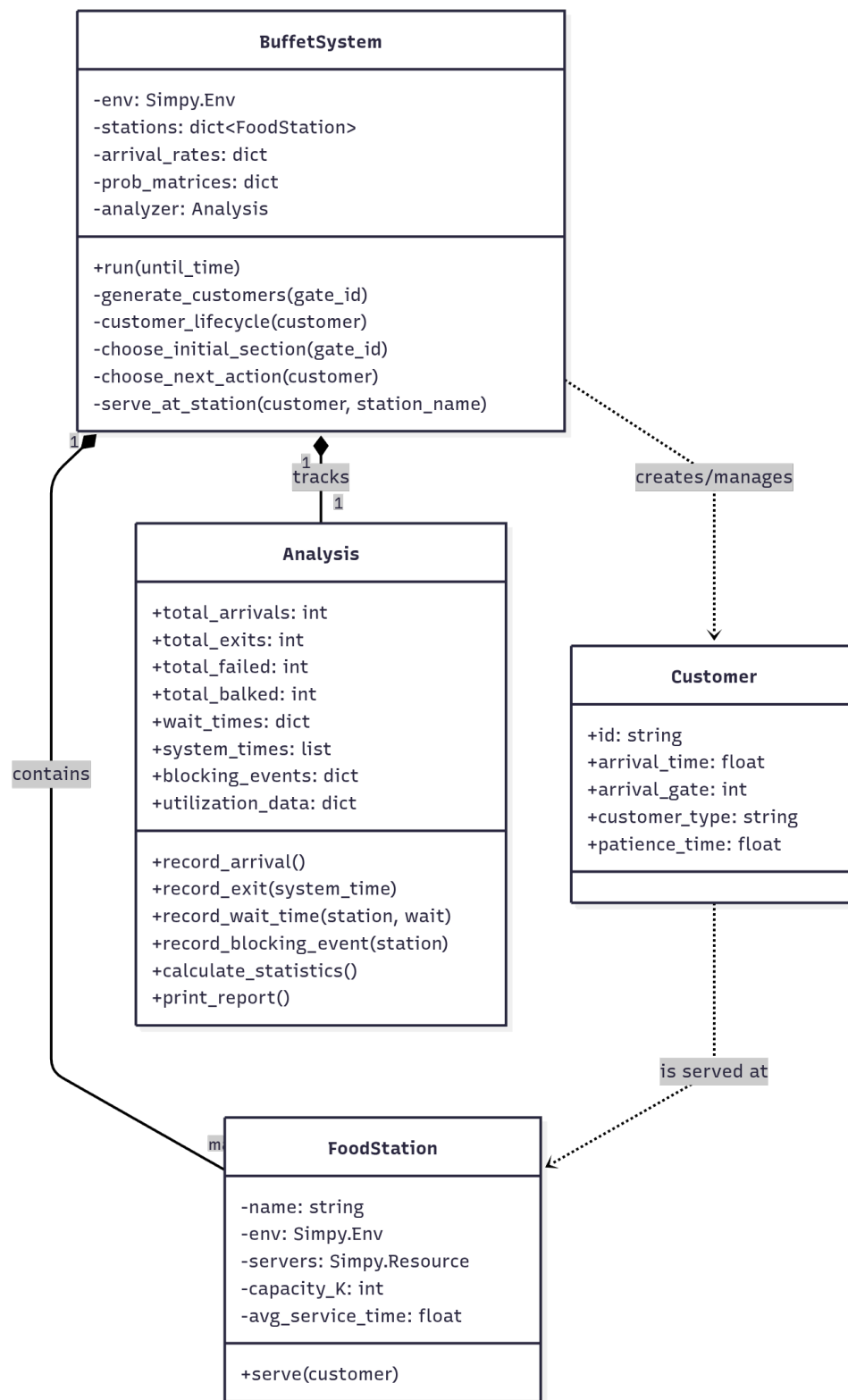
#### 3.1 Bản thiết kế mô hình hệ thống



Hình 2: Thiết kế mô hình hệ thống.

#### 3.2 Mô tả các module và sự tương tác giữa chúng

Sơ đồ khối các module



Hình 3: Mô tả các module.

## Giải thích chức năng

### 3.2.1 Lớp BuffetSystem (Hệ thống Nhà hàng)

- Đây là bộ não của toàn bộ mô phỏng. Nó chứa logic chính, điều khiển luồng thời gian và quản lý tất cả các thành phần khác.
- Thuộc tính chính:
  - env: Môi trường mô phỏng SimPy, quản lý thời gian và sự kiện.
  - stations: Một dictionary lưu trữ tất cả các đối tượng FoodStation để dễ dàng truy cập.
  - arrival\_rates: Lưu trữ tốc độ khách đến ( $\lambda_0, \lambda_1$ ).
  - prob\_matrices: Chứa các ma trận xác suất di chuyển
  - analyzer: Một đối tượng của lớp Analysis để ghi nhận mọi số liệu.
- Phương thức chính:
  - run(util\_time): Phương thức khởi động. Nó sẽ thiết lập các tiến trình ban đầu (như generate\_customers) và sau đó ra lệnh cho env chạy mô phỏng cho đến một mốc thời gian định trước.
  - generate\_customers(): Một "tiến trình" SimPy chạy song song. Nó chứa một vòng lặp vô tận, trong mỗi vòng lặp, nó sẽ yield một khoảng thời gian chờ ngẫu nhiên (dựa trên arrival\_rates) và sau đó tạo ra một đối tượng Customer mới.
  - serve\_customer(): Nó định nghĩa toàn bộ hành trình của một khách hàng, cụ thể nó nhận một khách hàng, tìm đối tượng FoodStation tương ứng và sau đó yêu cầu quầy đó phục vụ khách hàng.
  - Cụ thể như sau: 3 child method
    - \* random\_choose(): chọn quầy thức ăn ngẫu nhiên.
    - \* shortest\_expected\_wait(): khách hàng quan sát các quầy thức ăn và đi vào quầy hiện có ít người nhất.
    - \* one\_liner(): Toàn bộ khách hàng xếp thành một hàng dài đi qua lần lượt các quầy thức ăn, sau khi đi qua hết thì exit. Như vậy trực tiếp loại bỏ việc dựa vào xác suất đầu và không xảy ra việc quay lại lấy thức ăn ở quầy khác (vì đã đi qua tất cả rồi).

### 3.2.2 Lớp FoodStation (Quầy Thức ăn)

- Vai trò: Đại diện cho một hàng đợi M/M/c/K vật lý. Đây là nơi diễn ra hoạt động xếp hàng và "tự phục vụ".
- Thuộc tính chính:
  - servers: Đây là thành phần cốt lõi. Đối tượng này của SimPy chính là hiện thực hóa của hàng đợi M/M/c. capacity của nó chính là c (số kẹp gấp/muỗng). Nó tự động quản lý một hàng đợi nội bộ được sinh ra bởi serve().
  - capacity\_K: Một số nguyên đại diện cho giới hạn không gian vật lý. Logic kiểm tra K phải được thực hiện trước khi khách hàng cố gắng request() một server.
  - avg\_service\_time: Thời gian lấy thức ăn trung bình ( $1/\mu$ ), được dùng làm tham số cho hàm phân phối mũ để tạo ra thời gian phục vụ ngẫu nhiên cho mỗi khách.
- Phương thức chính:
  - serve(): Chứa logic cụ thể của việc yêu cầu server, chờ đợi, và giải phóng server. Cụ thể như sau:
    - \* FCFS(): Đây là cơ chế phục vụ tiêu chuẩn, đảm bảo khách hàng nào đến hàng đợi trước sẽ được phục vụ trước, được tích hợp sẵn và tối ưu hóa thông qua đối tượng simpy.Resource trong Simpy.



### 3.2.3 Lớp Customer (Khách hàng)

- Vai trò: Đại diện cho một "thực thể"(entity) di chuyển trong hệ thống. Nó chủ yếu là một cấu trúc dữ liệu để lưu trữ trạng thái và thuộc tính của từng khách hàng.
- Thuộc tính chính:
  - arrival\_gate: Ghi nhận khách hàng đến từ cổng 0 hay 1.
  - arrival\_time: Ghi nhận thời gian khách hàng xuất hiện.
  - serve\_time: bộ 4 giá trị thời gian khách hàng dành ra để lấy thức ăn ứng với mỗi quầy hàng(ngẫu nhiên, cố định với mỗi khách hàng).
  - Id: Mã định danh duy nhất để dễ dàng theo dõi và gỡ lỗi.
  - customer\_type: 'normal', 'indulgent', 'impatient', 'erratic': Một chuỗi ký tự ('normal', 'indulgent', 'impatient', 'erratic'). Thuộc tính này cho phép các hàm trong BuffetSystem áp dụng các logic khác nhau cho các loại khách hàng khác nhau (nhân đôi serve\_time cho 'indulgent', patience time thấp cho impatient, tăng thêm một lượng service\_time bằng nhau cho customer đứng sau erratic).
  - patience\_time: Ngưỡng kiên nhẫn để hiện thực hóa hành vi "Reneging"(rời hàng đợi).

### 3.2.4 **Lớp Analysis** (Bộ Phân tích)

- Vai trò: Tách biệt hoàn toàn logic thu thập và xử lý số liệu ra khỏi logic mô phỏng.
- Thuộc tính chính: Lưu trữ tất cả dữ liệu thô được thu thập trong quá trình mô phỏng (tổng số khách, thời gian chờ, sự kiện bị chặn...).
- Phương thức chính:
  - record\_...(): Các hàm được gọi bởi BuffetSystem tại các thời điểm quan trọng (khách đến, khách rời đi, bắt đầu chờ, khách bỏ về ...) để ghi nhận dữ liệu.
  - calculate\_statistics(): Phương thức này được gọi sau khi env.run() đã hoàn thành. Nó sẽ duyệt qua tất cả dữ liệu thô đã thu thập và tính toán các chỉ số có ý nghĩa (giá trị trung bình, độ lệch chuẩn, tỷ lệ phần trăm...).
  - print\_report(): Định dạng và in các kết quả đã được tính toán ra màn hình.

### 3.2.5 Mối Quan Hệ (Relationships)

- BuffetSystem "sở hữu" các FoodStation và một Analysis. Các thành phần này không thể tồn tại độc lập bên ngoài hệ thống.
- BuffetSystem tạo ra và quản lý các đối tượng Customer.
- Customer được gửi đến các FoodStation để được phục vụ.

## 4 Các thành phần chính của mô hình

Mô hình được xây dựng dựa trên các thành phần cốt lõi sau:

- Thực khách (Customer Agents): Khách hàng không phải là các thực thể thụ động, mà là các "tác nhân"(agents) có hành vi riêng. Mô hình định nghĩa các loại khách hàng khác nhau (ví dụ: người ăn nhanh, người ăn nhiều, người thiếu kiên nhẫn, người bất lịch sự) để tăng tính thực tế.
- Lối vào Đa luồng (Multi-Source Entrances): Hệ thống có hai lối vào riêng biệt (Arrived 0 và Arrived 1). Điểm đặc biệt là luồng khách từ mỗi lối vào có một "hồ sơ hành vi" khác nhau, thể hiện qua các xác suất lựa chọn quầy thức ăn ban đầu khác nhau.
- Các Quầy Thức ăn (Food Stations as Queues): Mỗi quầy thức ăn (Thịt, Hải sản, Tráng miệng...) được mô hình hóa chính xác là một hàng đợi M/M/c/K.

- $c$  (Số máy chủ) được ánh xạ trực tiếp vào các vật dụng vật lý như "sổ kẹp gấp"(Tongs) hoặc "sổ múỗng múc"(Ladle), đại diện cho số người có thể lấy thức ăn đồng thời.
- $K$  (Sức chứa) đại diện cho giới hạn không gian vật lý tối đa tại mỗi khu vực quầy.
- Trạng thái Ra Quyết định (Decision Nodes): Các nút hình kim cương trong sơ đồ luồng là nơi khách hàng đưa ra quyết định dựa trên các xác suất được định nghĩa trước (ví dụ: "chọn quầy nào?" hoặc "lấy thêm đồ hay ra về?").

## 5 Luồng hoạt động cốt lõi

- Đến (Arrival): Khách hàng được tạo ra tại một trong hai lối vào ( $\lambda_0$  hoặc  $\lambda_1$ ) theo một quy luật ngẫu nhiên (phân phối Poisson).
- Lựa chọn Ban đầu: Ngay lập tức, khách hàng ra quyết định đầu tiên: chọn đi đến quầy thức ăn nào dựa trên ma trận xác suất tương ứng với cổng vào của họ.
- Phục vụ tại Quầy: Khách hàng di chuyển đến quầy đã chọn. Nếu quầy còn chỗ ( $< K$ ) và còn "máy chủ" trống quầy thức ăn còn chỗ gấp ( $< c$ ), họ sẽ lấy thức ăn ngay. Nếu không, họ sẽ xếp hàng. Nếu hàng chờ của một quầy đã đầy khách hàng sẽ không thể đến quầy đó nữa. Nếu tất cả các quầy đều đã đầy thì khách hàng đến sẽ bỏ về (fail).
- Lựa chọn tiếp theo: Sau khi lấy xong thức ăn tại một quầy, họ quay lại một điểm quyết định chung. Tại đây, họ lại dùng xác suất để quyết định: (a) đi lấy thêm đồ ăn ở một quầy khác (có thể là quầy cũ hoặc quầy mới mỗi quầy chỉ được đi qua tối đa một lần - đơn giản hóa cho `customer_type = indulgent`) hoặc (b) quyết định ra về.
- Lặp lại hoặc Rời đi: Nếu chọn lấy thêm đồ, chu trình 2-3-4 sẽ lặp lại. Nếu chọn ra về, họ sẽ di chuyển đến điểm "Exit" và bị xóa khỏi hệ thống.