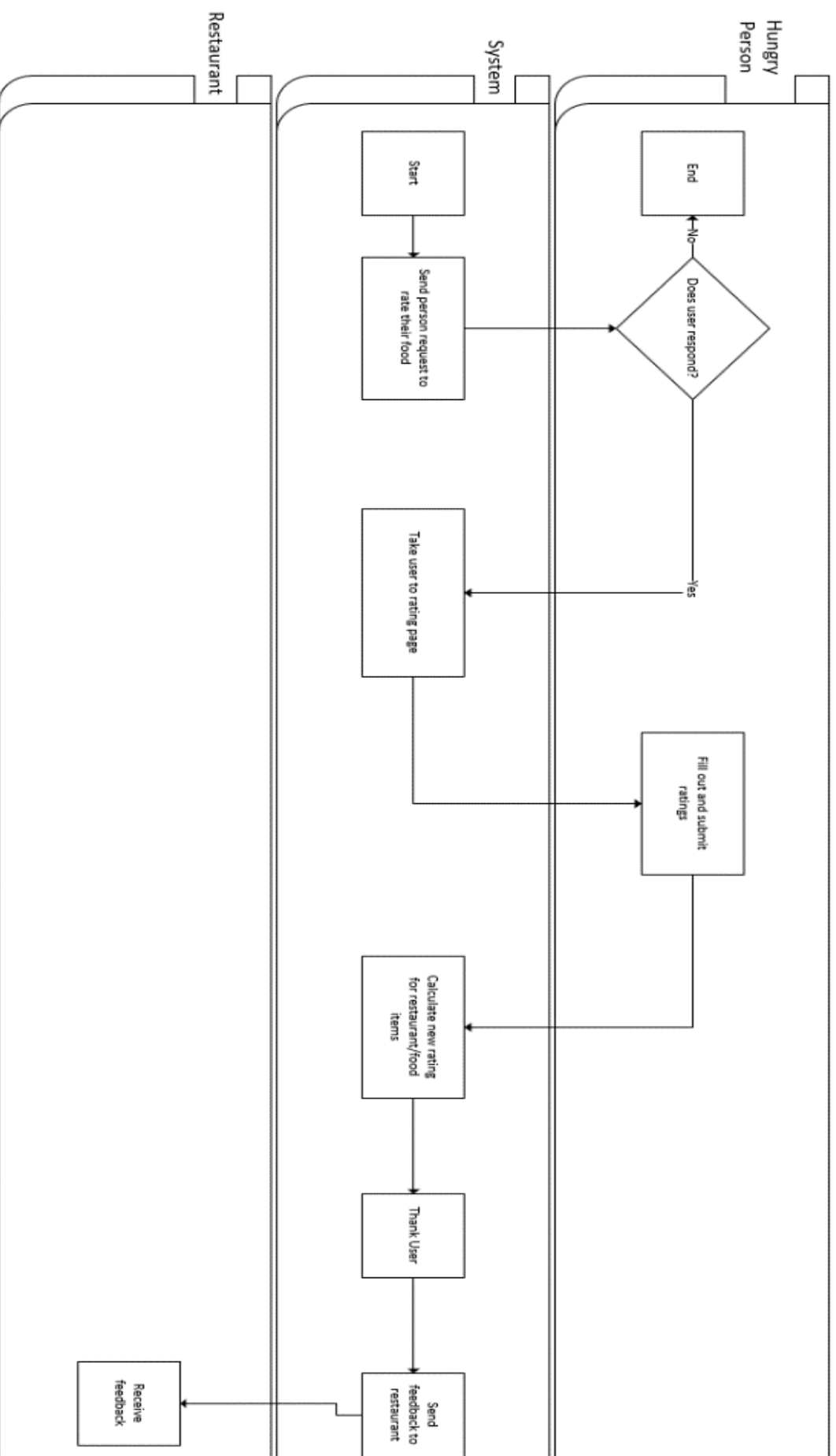Eric Murphy

Project Part 2 – Individual

- **Project:** Eat-Net
- **Requirements:**
  - UR-9, Rate restaurants and items
  - UR-8, Being able to view orders from consumers
- **Use Case Documents:**
  -

Use Case ID:	UR-8		
Use Case Name:	View costumer orders		
Description:	Allow Restaurants to view orders from customers		
Actors:	Restaurants		
Precondtions:	Customer has placed food order, Customer has payed for food order		
Postconditions:	Customer recieves food		
Frequency of Use:	After every food order		
Flow of Events:		Actor Action	System Response
	1		Notify Restaurant of new food order
	2	Business makes the food	
	3	Customer recieves the food	
Variations:	None		
Exceptions:	Invalid Order		
Developer Notes:	None		

  -

Use Case ID:	UR-9		
Use Case Name:	Rate Restaurants and Items		
Description:	Users can rate restaurants and food items after they receive their order		
Actors:	Hungry People		
Precondtions:	Food order has been received		
Postconditions:	Rating is calculated into the total rating		
Frequency of Use:	After every food order		
Flow of Events:		Actor Action	System Response
	1		Sends Email to user asking for a rating
	2	User goes to rate the restaurant and/or food items	
	3	User fills out ratings	
	4		Process the ratings
	5		Calculate the new rating for the restaurant/food item
	6		Thank the user
Variations:	None		
Exceptions:	Invalid Rating		
Developer Notes:	None		

- **Activity Diagram:**

UR-9, Rate a Restaurant/Food Item (Eric Murphy)

Hungry Person

End

Does user respond?

— No →

— Yes →

Fill out and submit ratings

System

Start

Send person request to rate their food

Take user to rating page

Calculate new rating for restaurant/food items

Thank User

Send feedback to restaurant

Restaurant

Receive feedback

UR-8, Rate a restaurant/food item (Eric Murphy)