

Analysis Document: Car Rental Service



Group 4

CSCI 4050: Software Engineering

Instructor: Krys Kochut

Osama Mansour, Stephen Patton, Vincent Lee, and Minh Pham

Table of Contents

2. Proposed System	x
2.1 System Model	x
2.1.1 Object Model	
2.1.1.1 Data Dictionary	
2.1.1.1.1 Entity Package	
2.1.1.1.2 user.control package..	
2.1.1.1.3 user.boundary package.....	
2.1.1.1.4 administrator.control package.....	
2.1.1.1.5 administrator.boundary package.....	
2.1.1.1.6 customer.control package	
2.1.1.1.7 customer.boundary package.....	
2.1.1.2 Class Diagrams	
2.1.1.2.1 Package Diagram	
2.1.1.2.2 Entity Package.	
2.1.1.2.3 user.control package.	
2.1.1.2.4 user.boundary package.....	
2.1.1.2.5 administrator.control package.....	
2.1.1.2.6 administrator.boundary package	
2.1.1.2.7 customer.control package..	
2.1.1.2.8 customer.boundary package.....	
2.1.2 Dynamic Model – Sequence & Communication Diagrams.....	
2.1.2.1 ReturnVehicle.	
2.1.2.2 RegisterCustomer.....	
2.1.2.3 AddVehicle.	
2.1.2.4 ListVehicle..	26
2.1.2.5 ModifyVehicle.	28
2.1.2.6 DeleteVehicle.	29
2.1.2.7 AddVehicleType.	31
2.1.2.8 ListVehicleType.	33
2.1.2.9 ModifyVehicleType.	35
2.1.2.10 DeleteVehicleType.	37
2.1.2.11 ListProfile.	
2.1.2.12 ModifyProfile	41
2.1.2.13 DeleteProfile	43
2.1.2.14 Login	
2.1.2.15 LogOut	47
2.1.2.16 AddUser	49
2.1.2.17 ListUser	51
2.1.2.18 ModifyUser	53
2.1.2.19 TerminateMembership.....	
2.1.2.20 CancelReservation	57
2.1.2.21 ReserveCar	59
3. Glossary	61

2.1.1 Object Model

2.1.1.1 Data Dictionary

2.1.1.1.1 Entity Package

Class Name	User		
Purpose	The generalize entity involved in a Car rental system transaction		
SuperClass	None		
SubClass	Administrator, Customer		
Attributes	Name	Type	Description
	UserID	String	The ID of the User as defined by the System Registration
	password	String	The encryption on the user profile to be accessed
	role	String	The role of the user defined by the system
Operations	Name	Inputs	Description
	Login	UserID, Password role	The login page allows users access to registered users within a system
	Logout		The logout page allows users currently login in to exit there current system profile
	updateProfile		The update profile operation allows for user to change and save their profile information
Relationships	Super class of Administrator, Customer		

Analysis Document: Car Rental Service 4

Class Name	DriversLicense		
Purpose	Provide driver's license information of Customer		
SuperClass	None		
SubClass	None		
Attributes	Name	Type	Description
	number	String	The number on the license
	expiration	String	The expiration date
	dateOfBirth	String	The date of birth on the license
Operations	Name	Inputs	Description
Relationships	None		

Class Name	CreditCard		
Purpose	Provide credit card information of Customer		
SuperClass	None		
SubClass	None		
Attributes	Name	Type	Description
	number	String	The number on the card
	expiration	String	The expiration date
	code	Int	The CVC code
Operations	Name	Inputs	Description
Relationships	None		

Analysis Document: Car Rental Service 5

Class Name	Customer			
Purpose	The generalization of group of Users labeled Customers			
SuperClass	Users			
SubClass	None			
Attributes	Name	Type	Description	
	Social Security Number	String	Customer social security information for verification	
	firstName	String	Customer First name	
	lastName	String	Customer Last name	
	address	String	Customer Address	
	telephone	String	Customer Telephone number	
Operations	Name	Inputs	Outputs	Description
	ViewMyInfo		Personal Information Display	Displays all personal information of the current logged in Customer
	ViewAvalibleCars	CarName CarLocation	Available Cars	Displays the Cars available for rental by location
	ViewMyRentalHistory		Rental History Report	Displays a current logged Customer rental history
	ViewMyBillingInfo		Billing Report	Displays Current logged Customer billing report
	ViewMyCarRental		Car Rental Report	Displays Customer past and present car rentals
Relationships	EditMyInfo	FirstName LastName Address Telephone #	Updated Personal Information	Allows Customers to edit their information as needed
	Subclass of the User			
	A Customer can only rent 1 Car at a time			
	A Customer can only edit personal information, but not vehicle information			

Analysis Document: Car Rental Service 6

Class Name	Administrator			
Purpose	The generalization of group of Users labeled Administrator within the Car rental System			
SuperClass	Users			
SubClass	None			
Attributes	Name	Type	Description	
	AdminID	Int	ID identifying the Administrator within the car rental system	
	firstName	String	First name	
	lastName	String	Last name	
	address	String	Address	
	telephone	String	Telephone number	
Operations	Name	Inputs	Outputs	Description
	ViewMyInfo		Personal Information Display	Displays all personal information of the current logged in administrator
	ViewVehicles	CarName CarLocation	Available Cars including model, make year, vin number, and repair history reports.	Displays the Cars available within the Car rental system
	ViewRentalHistory	Customer Name	Rental History Report on any Customer within the system	Displays a rental history for all customer by name
	ViewBillingHistory	Customer Name	Billing Report	Displays all Customer History Billing reports
	ViewCarRentals	Customer Name	Car Rental Report	Displays All Customer Car rental reports
	EditVehicleInfo	Vehicle Name	Updated Vehicle Information	Edit Vehicle Information
Relationships	EditCustomerInfo	CustomerName	Update Customer Personal Information	Allows Admin to edit customer information both adding and removing Customers
	Subclass of User			

Analysis Document: Car Rental Service 7

	An administrator can edit Vehicle Information, and Customer information on Request
--	--

Class Name	VehicleType			
Purpose	An entity class for storing information on vehicle types			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
	name	String	The name of the vehicle type	
	hourly_rate	Double	The hourly price of the vehicle type	
	daily_rate	Double	The daily price of the vehicle type	
Operations	Name	Inputs	Outputs	Description
Relationships	None			

Class Name	Vehicle			
Purpose	An entity class for storing information on vehicles in the fleet			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
	make	String	The make of the vehicle	
	model	String	The model of the vehicle	
	year	Integer	The manufacture year of the vehicle	
	tag	String	The registration tag of the vehicle	
	mileage	Integer	The current mileage of the vehicle	
	last_serviced	Date	The last date the vehicle was serviced	
	condition	String	The condition of the vehicle	
Operations	Name	Inputs	Outputs	Description
Relationships	A vehicle belongs to 1 location			

Analysis Document: Car Rental Service 8

Class Name	Location			
Purpose	An entity class for storing information on point of presence POP locations			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
	name	String	The name of the location	
	address	String	The address of the location	
	vehicle capacity	Integer	The capacity of the location	
	vehicles	ArrayList<Vehicle>	The list of vehicles assigned to the location	
Operations	Name	Inputs	Outputs	Description
Relationships	None			

Class Name	Membership			
Purpose	An entity class for storing information on membership tiers			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
	name	String	The name of the membership tier	
	monthly price	Double	The price of one month's service	
	duration	Integer	The length of membership in months	
Operations	Name	Inputs	Outputs	Description
Relationships	A customer belongs to 1 membership			

2.1.1.1.2 user.control package

Class Name	User			
Purpose	This control class allows a User to complete actions associated with login and profile modification			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	login	username password		Logs the user in with the appropriate permissions
	logout			Logs the user out
	loadProfile	username		Gets all relevant information related to the username
	updateProfile	profile	Boolean	Updates profile information and returns confirmation with Boolean
	registerCustomer	username password address creditcard driverlicense	Boolean	Allows a User to register as a customer
	vehicleSearch	name		Performs a search of vehicles in the fleet
Relationships	None			

2.1.1.1.3 user.boundary package

Class Name	LoginUI			
Purpose	This boundary class allows a User to log in			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	login	username password		Verify the user and logs the user with appropriate permissions
Relationships	None			

Class Name	LogoutUI			
Purpose	This boundary class allows a User to log out			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	logout			Logs the user out
Relationships	None			

Class Name	ChangeProfileUI			
Purpose	This boundary class allows a User to perform profile changes			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	updateProfile	profile		Updates the profile of a user
	saveProfile	profile	Boolean	Saves changes of profile change to data structure
Relationships	None			

Class Name	RegisterUI			
Purpose	This boundary class allows a User to register as a customer			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	registerCustomer	username password address creditcard driverlicense	Boolean	Registers a customer
Relationships	None			

Class Name	SearchUI			
Purpose	This boundary class allows a User to search the vehicle fleet			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	serach	keyword	List	Gets a list of available vehicles based on keyword
Relationships	None			

2.1.1.1.4 administrator.control package

Class Name	UserAdd			
Purpose	This control class allows Administrator to add users of all permissions			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	addUser	username password	Boolean	Adds a user to the database with Boolean confirmation
Relationships	None			

Class Name	VehicleRemove			
Purpose	This control class allows Administrator to remove a vehicle from the system			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	removeVehicle	vehicleProfile	Boolean	Removes a vehicle from the fleet with Boolean conformation
Relationships	None			

Class Name	VehicleTypeRemove			
Purpose	This control class allows a Administrator to remove a vehicle type from the system			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	removeVehicleType	vehicleType	Boolean	Removes a vehicletype from the database with confirmation
Relationships	None			

2.1.1.1.5 administrator.boundary package

Class Name	UserAddUI			
Purpose	This boundary class allows an Administrator to add a user to the domain			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	addUser	username password	Boolean	Adds a user to the system confirmation
Relationships	None			

Class Name	RemoveVehicleUI			
Purpose	This boundary class allows Administrator to remove a vehicle from the fleet			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	removeVehicle	vehicleProfile	Boolean	Removes a vehicle from the fleet with conformation
Relationships	None			

Class Name	RemoveVehicleTypeUI			
Purpose	This boundary class allows Administrator to remove a vehicletype from the system			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	removeVehicleType	vehicleType	Boolean	Removes a vehicletype from the site with confirmation
Relationships	None			

Class Name	VehicleTypeUI			
Purpose	This boundary class allows a Administrator to perform actions on the vehicle types			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	addVehicleType	name hourly_rate daily_rate	Boolean	Adds a vehicle type to the system with Boolean confirmation
	modifyVehicleType	vehicleType	Boolean	Modifies a vehicle type information with Boolean confirmation
Relationships	None			
Class Name	UserUI			
Purpose	This boundary class allows an Administrator to view information on users			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	listUser	keyword	List	Lists users based on keyword match
Relationships	None			

2.1.1.1.8 customer.control package

Class Name	Membership			
Purpose	This control class allows a Customer to perform actions to their membership options			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	terminateMembership		Boolean	Terminates a membership of a customer with Boolean confirmation
Relationships	None			

Class Name	VehicleReservation			
Purpose	This control class allows a Customer to preform actions related to renting a vehicle			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	reserveVehicle	vehicleProfile	Boolean	Reserves a vehicle with Boolean confirmation
	returnVehicle	vehicleProfile	Boolean	Checks a vehicle back into the fleet of available vehicles with Boolean confirmation
Relationships	None			

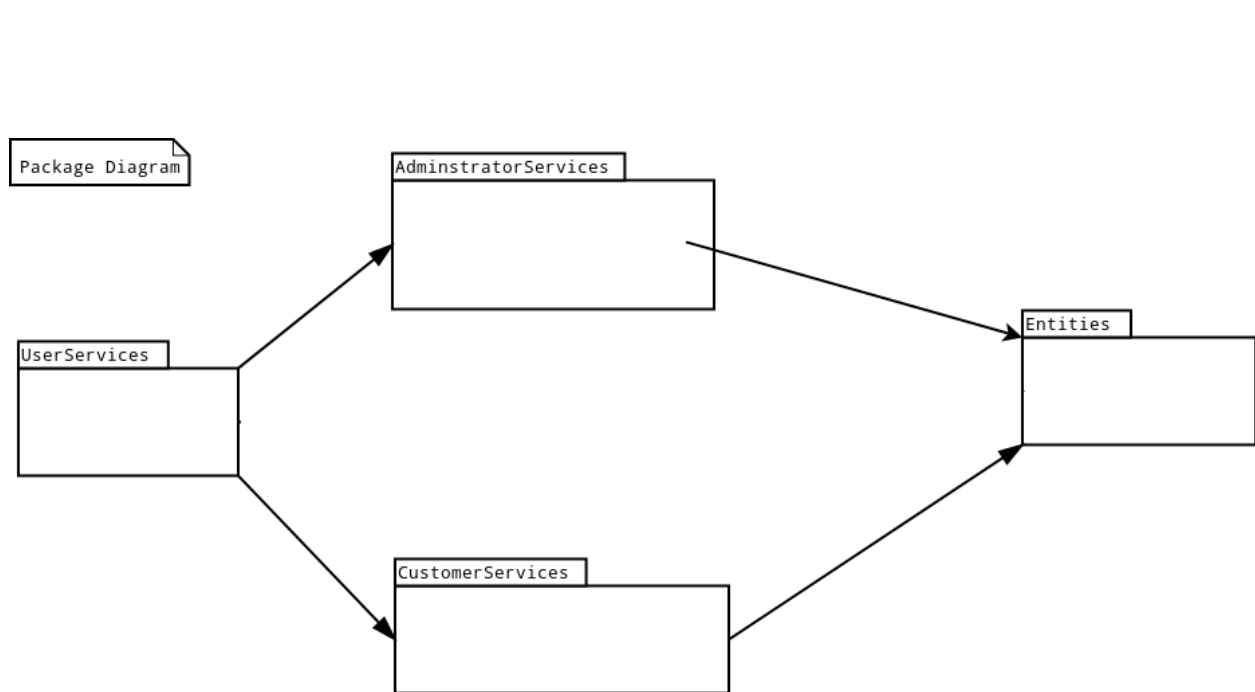
2.1.1.1.9 customer.boundary package

Class Name	MembershipUI			
Purpose	This boundary class allows a Customer to terminate their membership			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	terminateMembership		Boolean	Terminates a customer's membership with conformation
Relationships	None			

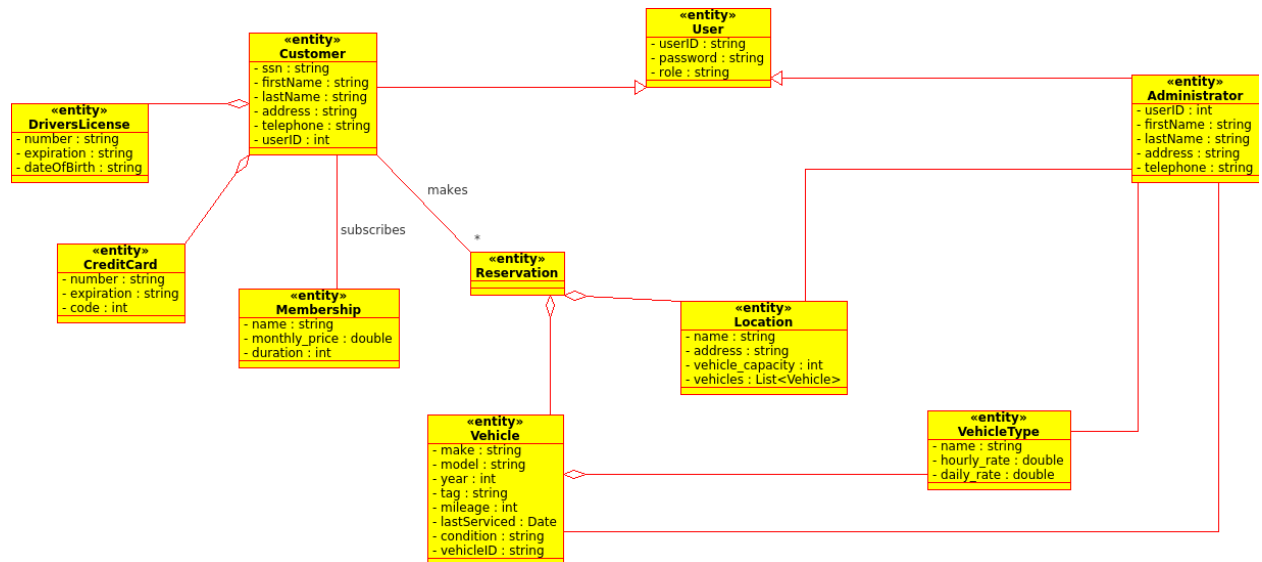
Class Name	ReservationUI			
Purpose	This boundary class allows a Customer to reserve a vehicle			
SuperClass	None			
SubClass	None			
Attributes	Name	Type	Description	
Operations	Name	Inputs	Outputs	Description
	reserveVehicle	vehicleProfile	Boolean	Reserves a vehicle with Boolean confirmation
	returnVehicle	vehicleProfile	Boolean	Checks a vehicle back into the fleet of available vehicles with Boolean confirmation
Relationships	None			

2.1.1.2 Class Diagrams

2.1.1.2.1 Package Diagram



2.1.1.2.2 Entity Package



2.1.1.2.3 user.control package

«control» User
+ login(username : string, password : string) + logout() + updateProfile(profile : profile) : bool + loadProfile(username : string) + registerCustomer(username : string, password : string, address : string, creditCard : CreditCard, driverlicense : DriverLicense) : bool + vehicleSearch(name : string)

2.1.1.2.4 user.boundary package

«boundary» LoginUI
+ login(username : string, password : string)

«boundary» logoutUI
+ logout()

«boundary» changeProfileUI
+ updateProfile(profile : Profile) + saveProfile(profile : Profile) : bool

«boundary» registerUI
+ registerCustomer(username : string, password : string, address : string, creditCard : CreditCard, driverLicense : DriverLicense) : bool

«boundary» searchUI
+ search(keyword : string)

2.1.1.2.5 administrator.control package

«control» UserAdd
+ addUser(username : string, password : string) : bool

«control» VehicleRemove
+ removeVehicle(vehicleProfile : Vehicle) : bool

«boundary» VehicleTypeRemove
+ removeVehicleType(vehicleType : VehicleType) : bool

2.1.1.2.6 administrator.boundary package

«boundary» UserAddUI
+ addUser(username : string, password : string) : bool

«boundary» RemoveVehicleUI
+ removeVehicle(vehicleProfile : Vehicle) : bool

«boundary» RemoveVehicleTypeUI
+ removeVehicleType(vehicleType : VehicleType) : bool

2.1.1.2.9 customer.control package

«control» Membership
+ terminateMembership() : bool

«control» VehicleReservation
+ reserveVehicle(vehicleProfile : Vehicle) : bool
+ returnVehicle(vehicleProfile : Vehicle) : bool

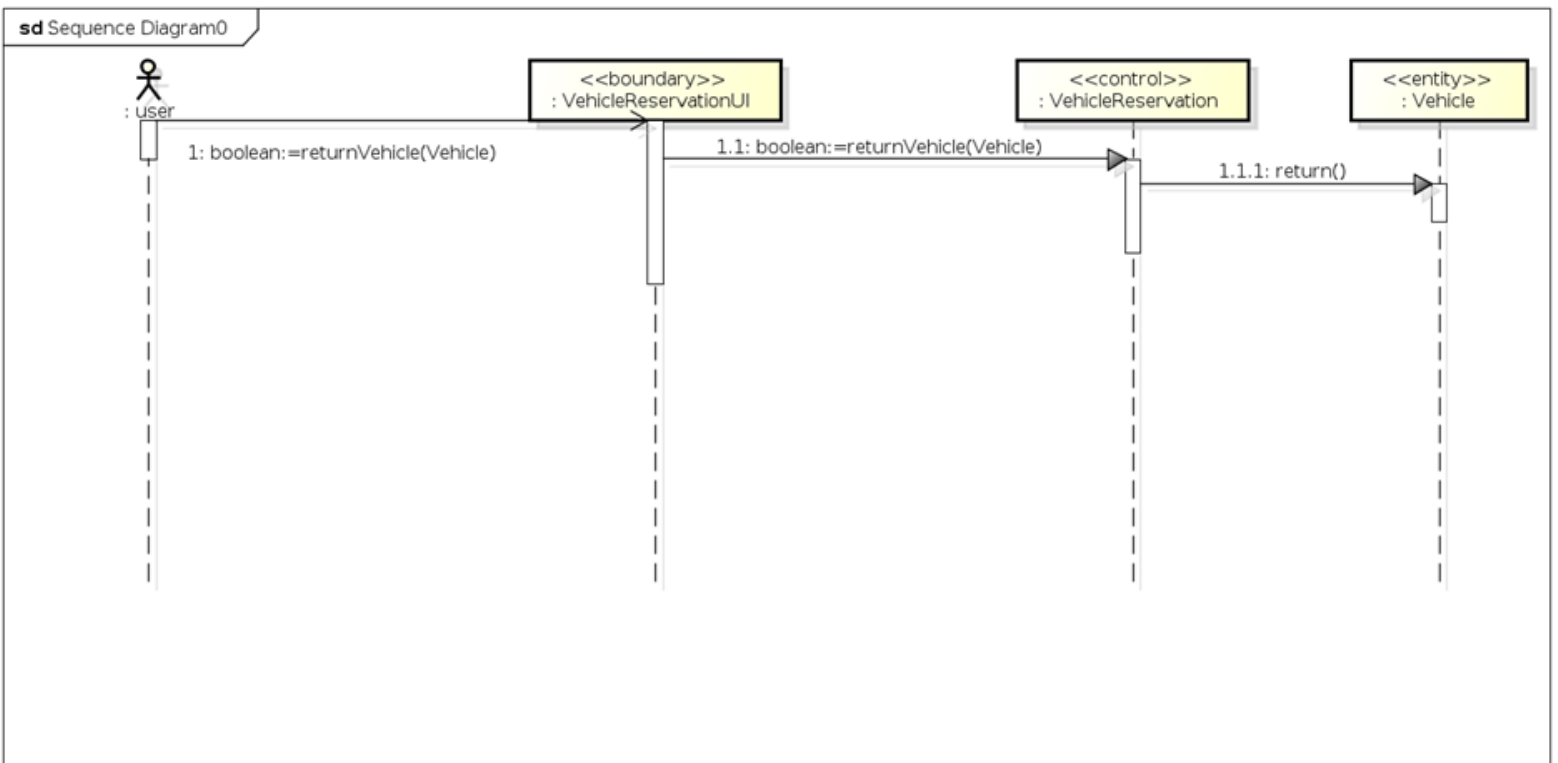
2.1.1.2.10 customer.boundary package

«boundary» MembershipUI
+ terminateMembership() : bool

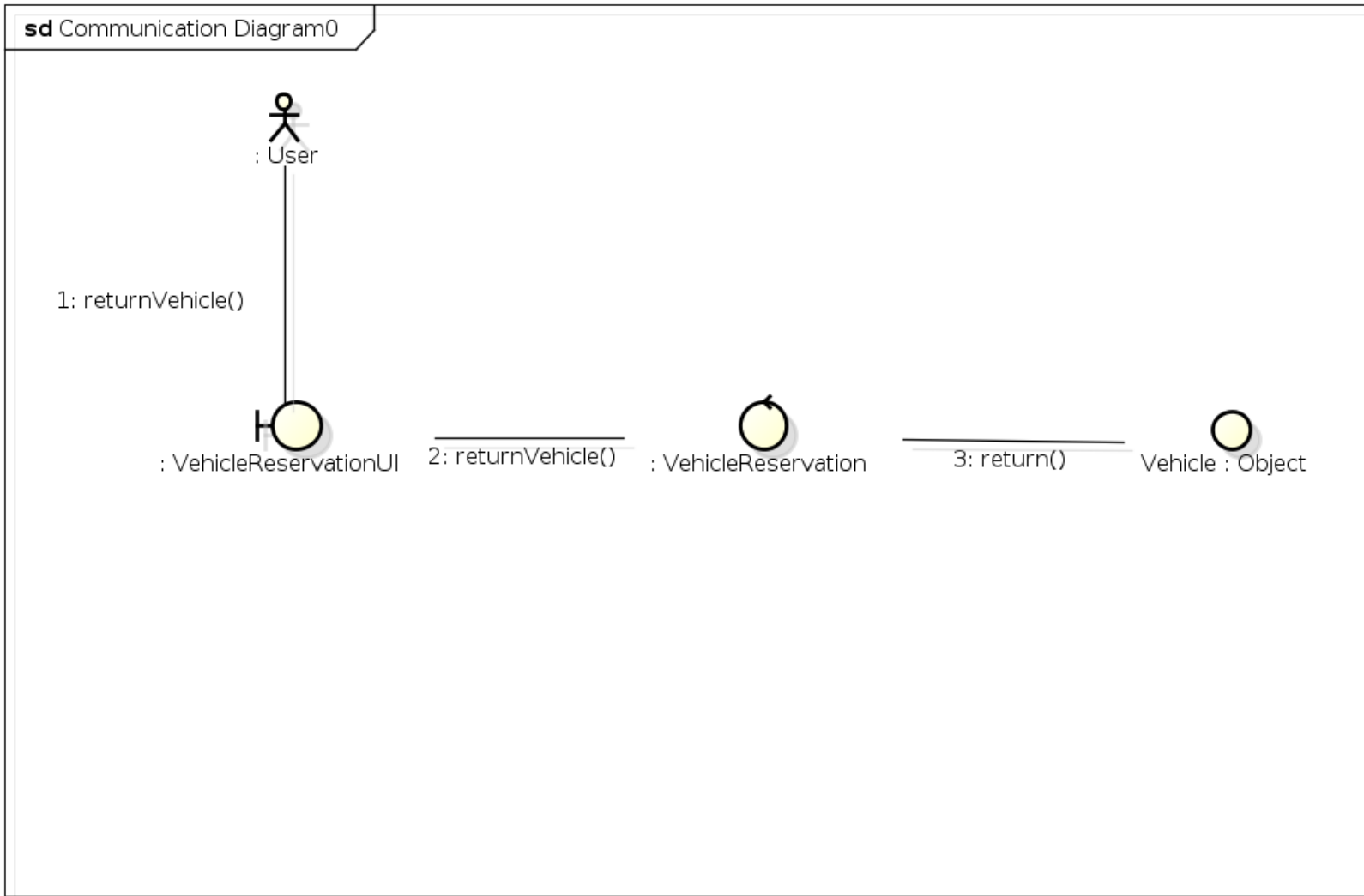
«boundary» VehicleReservationUI
+ reserveVehicle(vehicleProfile : Vehicle) : bool
+ returnVehicle(vehicleProfile : Vehicle) : bool

2.1.2 Dynamic Model – Sequence & Communication Diagrams

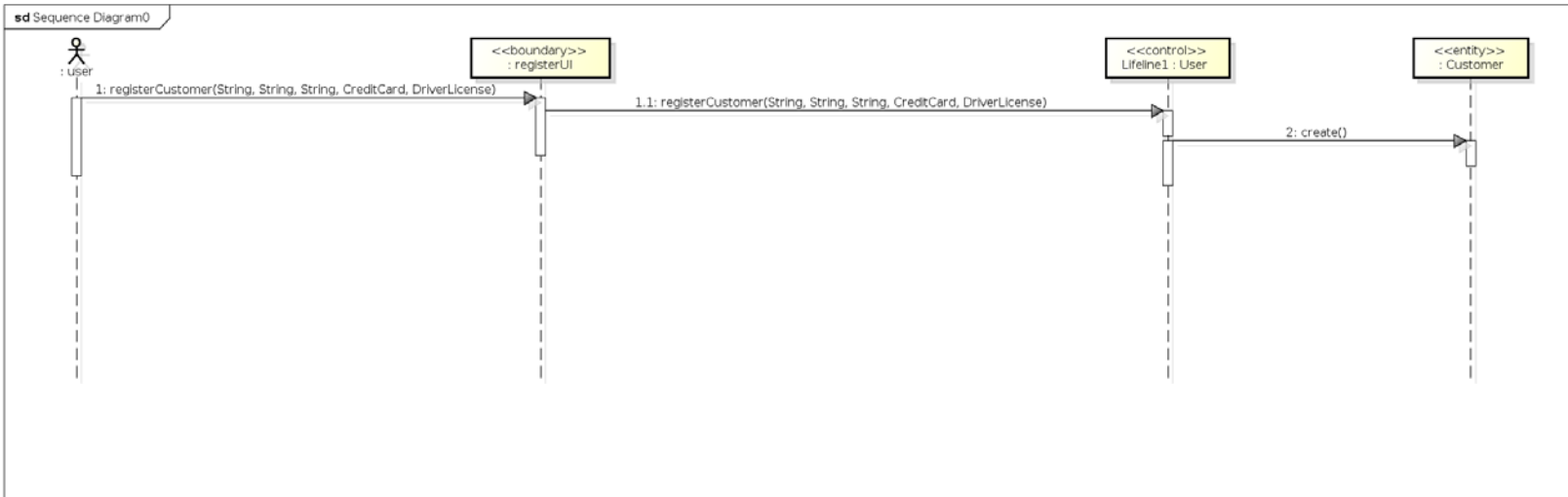
2.1.2.1 ReturnVehicle – Sequence Diagram



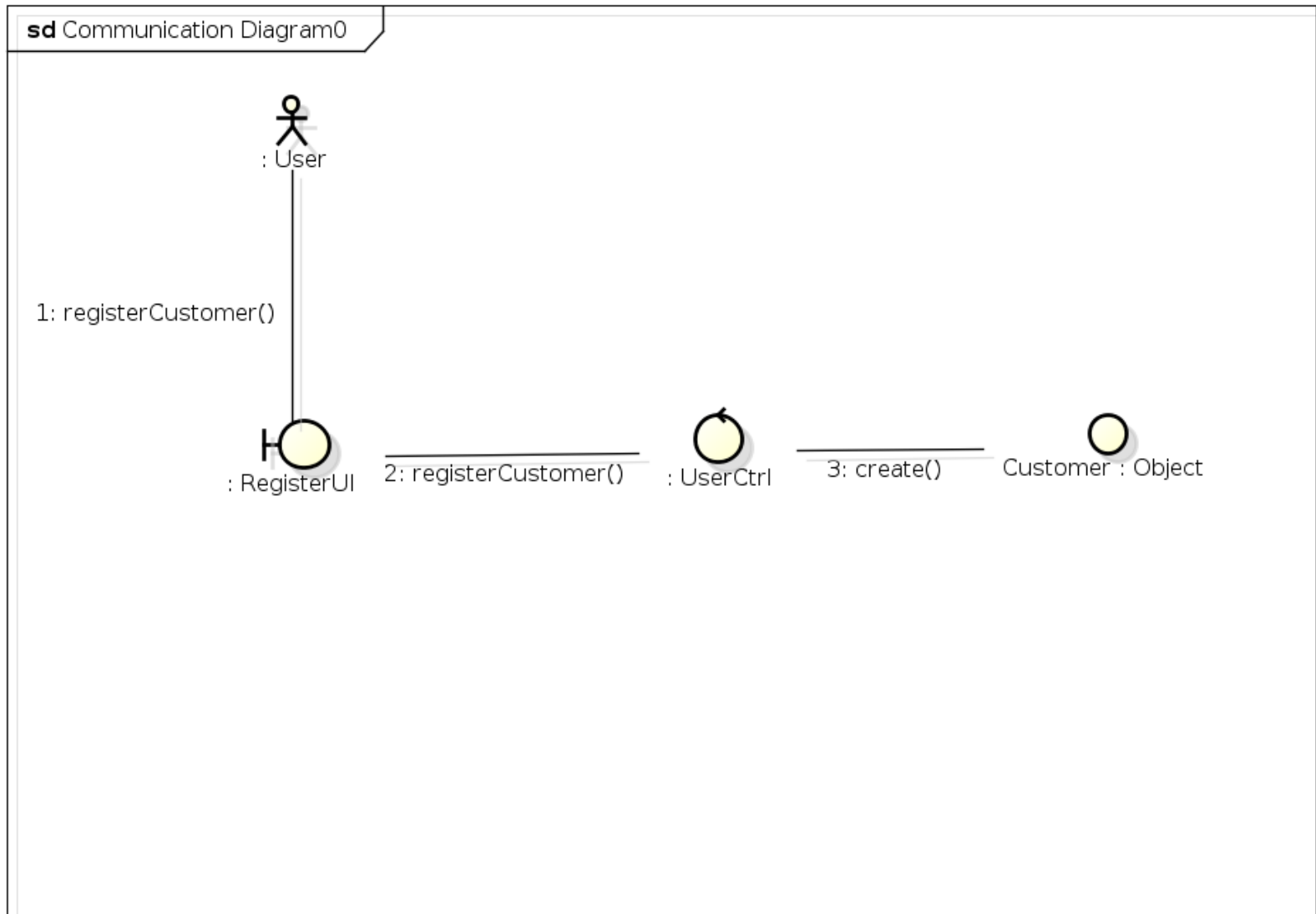
2.1.2.1 ReturnVehicle – Communication Diagram



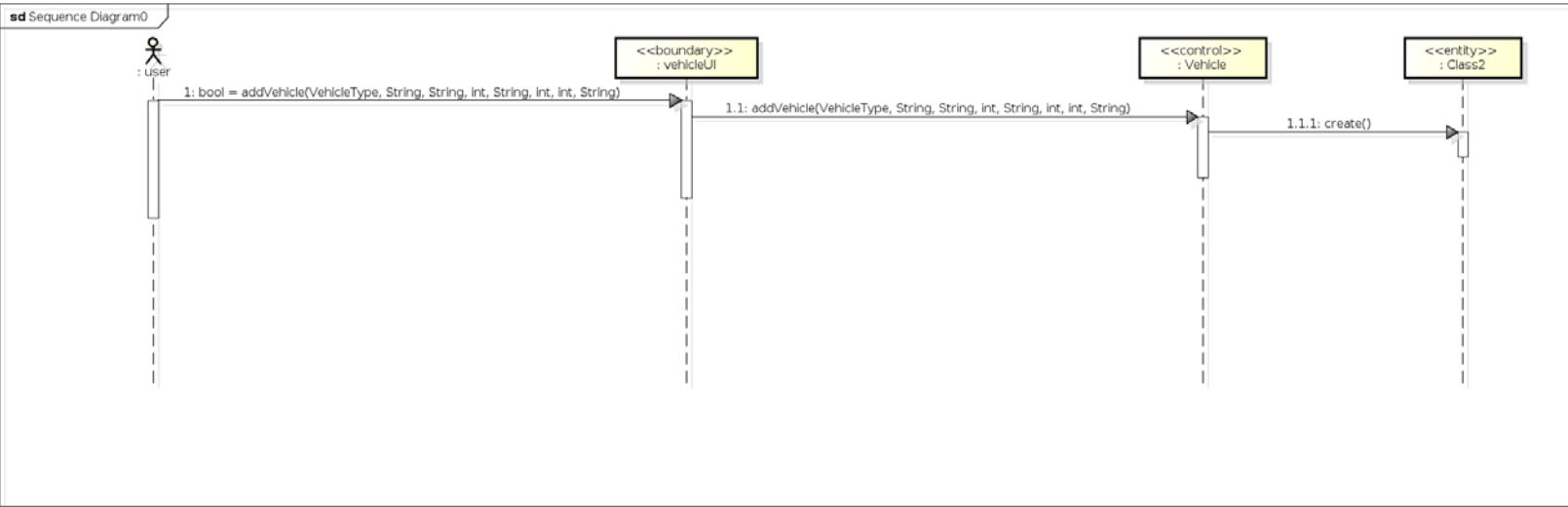
2.1.2.2 RegisterCustomer– Sequence Diagram



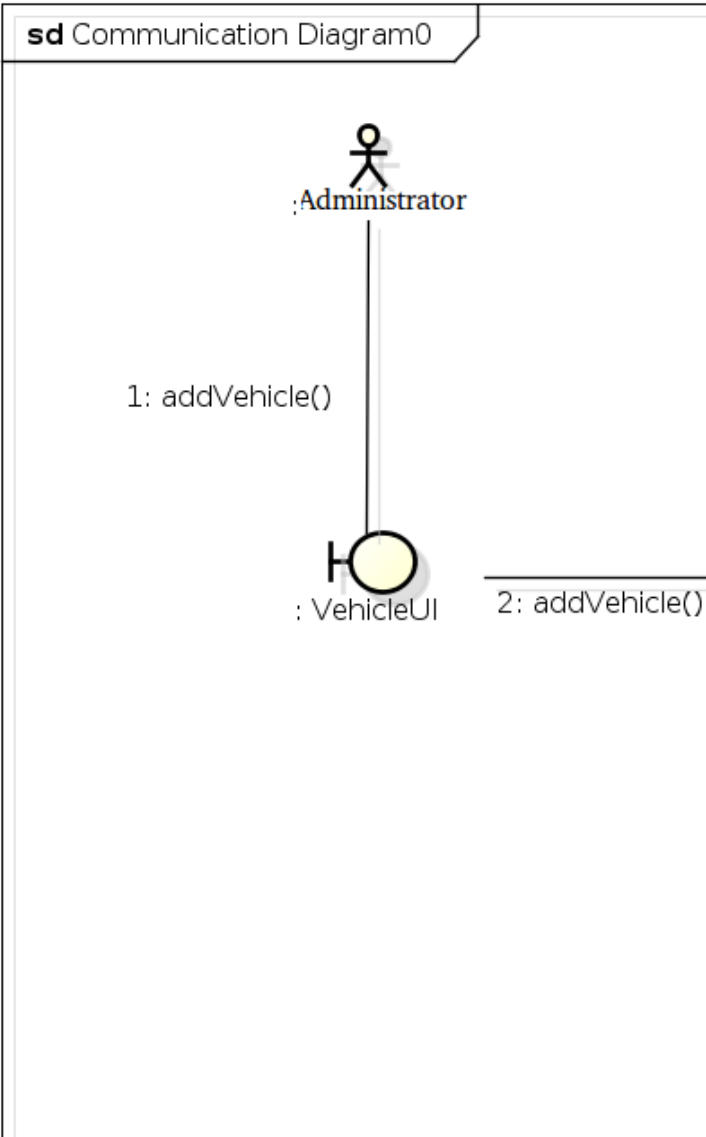
2.1.2.2 RegisterCustomer– Communication Diagram



2.1.2.3 AddVehicle – Sequence Diagram

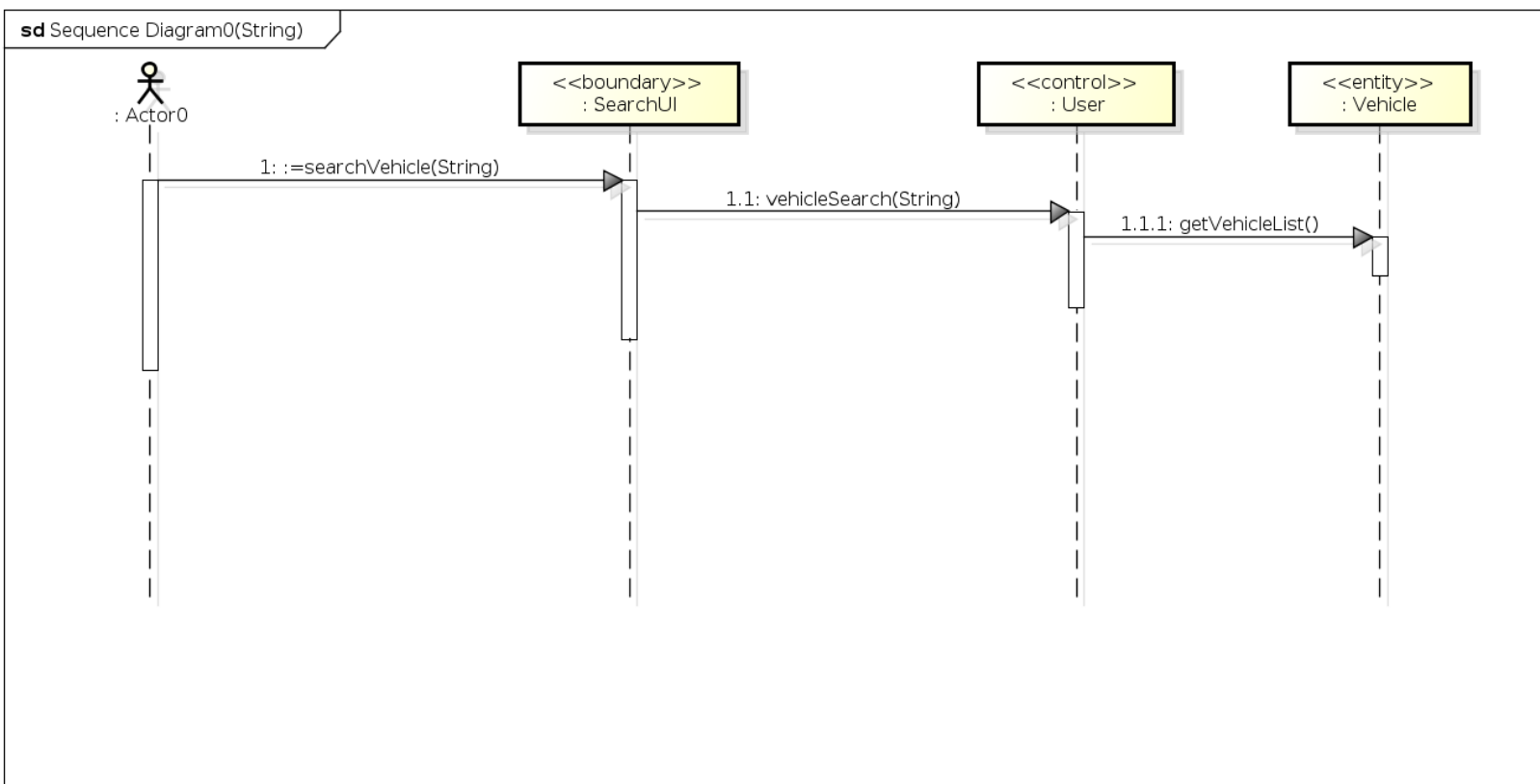


2.1.2.3 AddVehicle – Communication Diagram

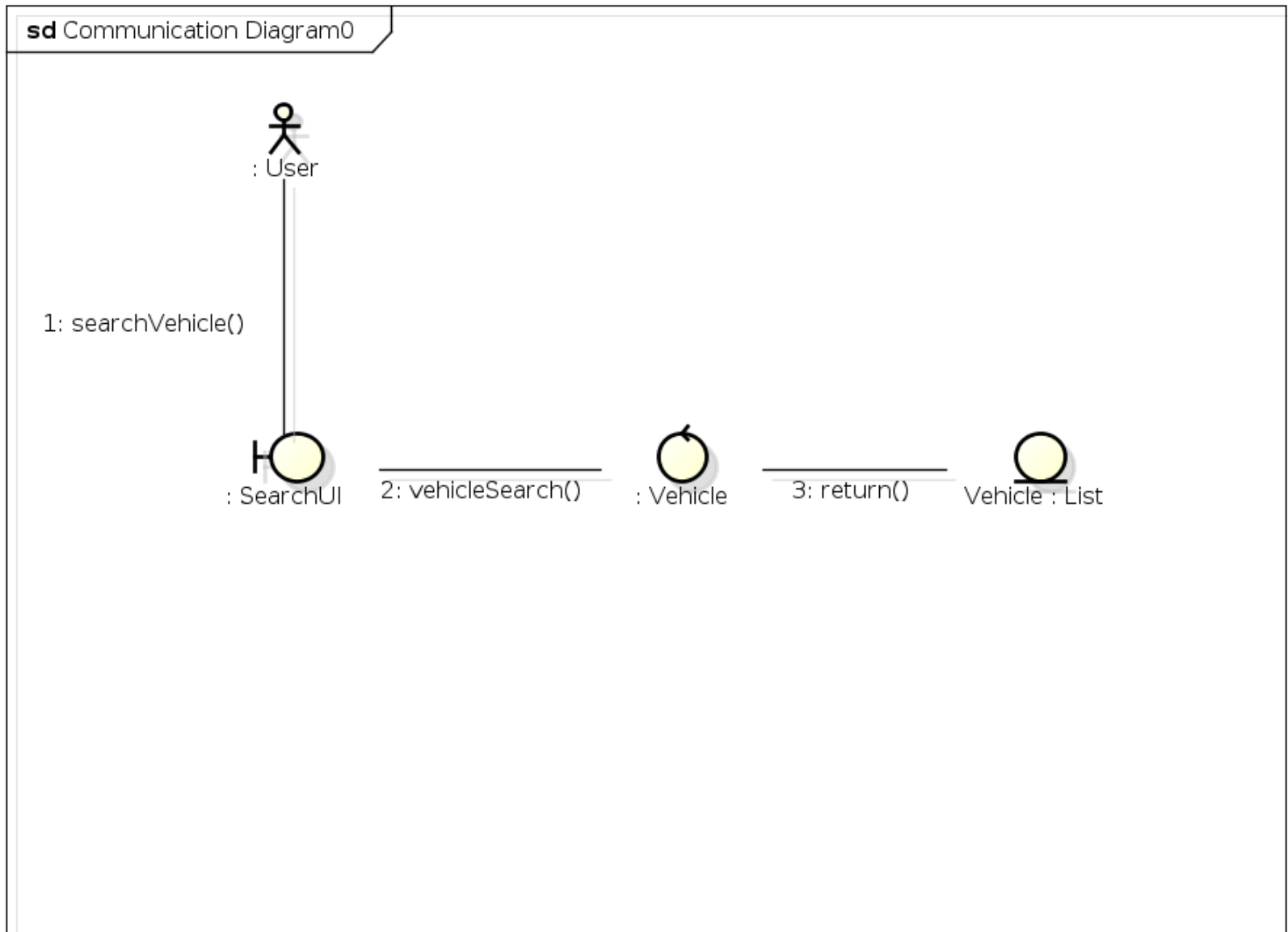


2.4.4.3 AddVehicle – Communication Diagram – Use

2.1.2.4 ListVehicle – Sequence Diagram

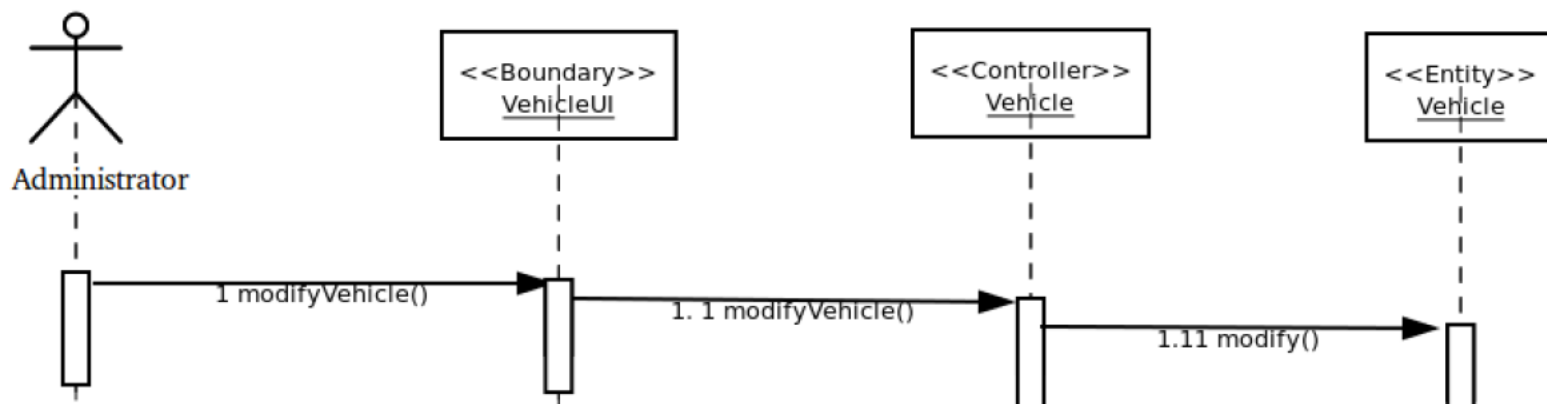


2.1.2.4 ListVehicle – Communication Diagram

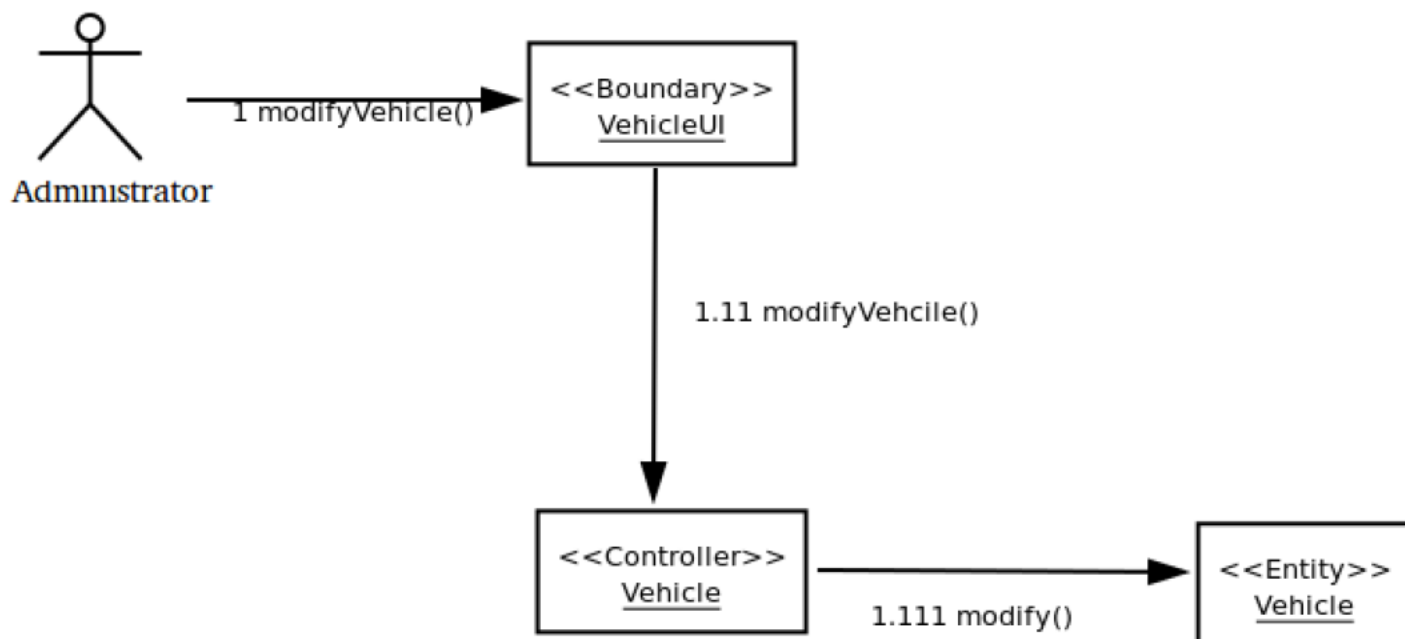


2.1.2.5 ModifyVehicle – Sequence Diagram

ModifyVehicle: Sequence Diagram

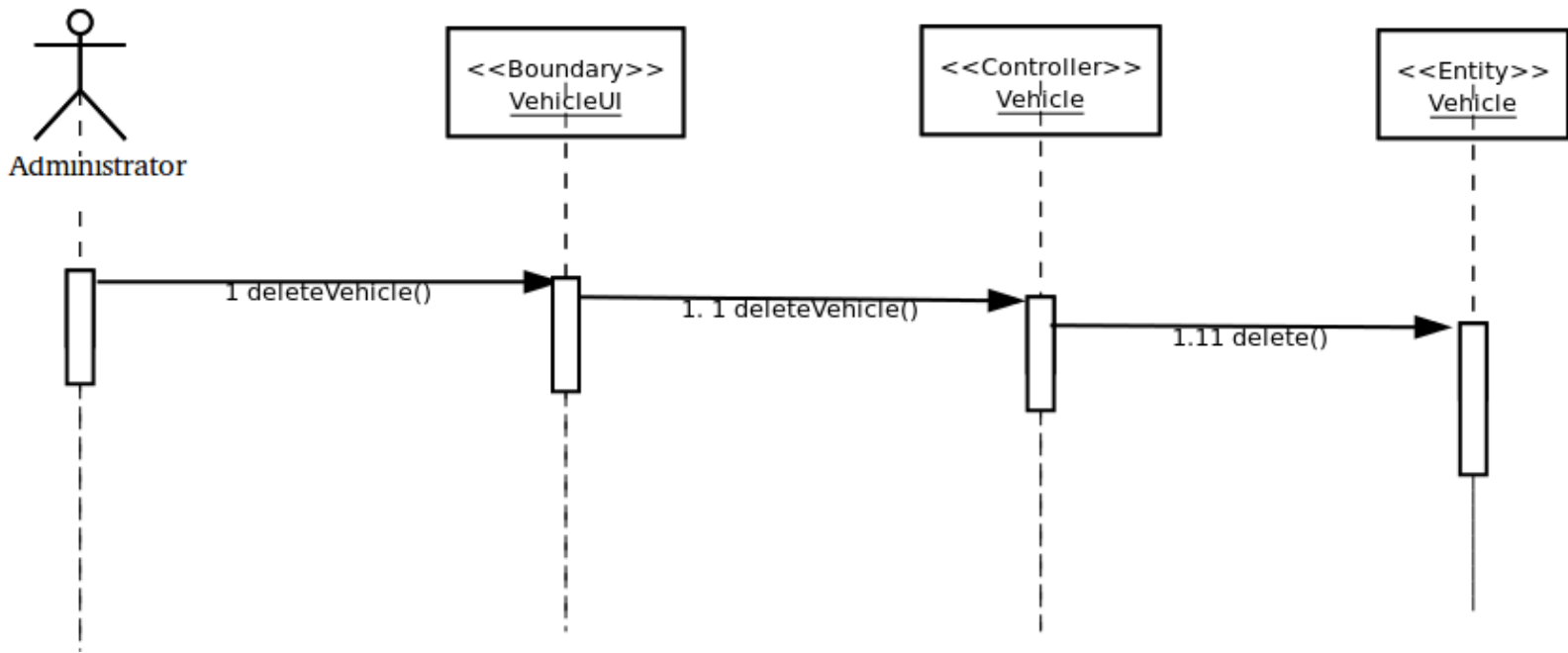


Modify Vehicle: Collobaration Diagrams



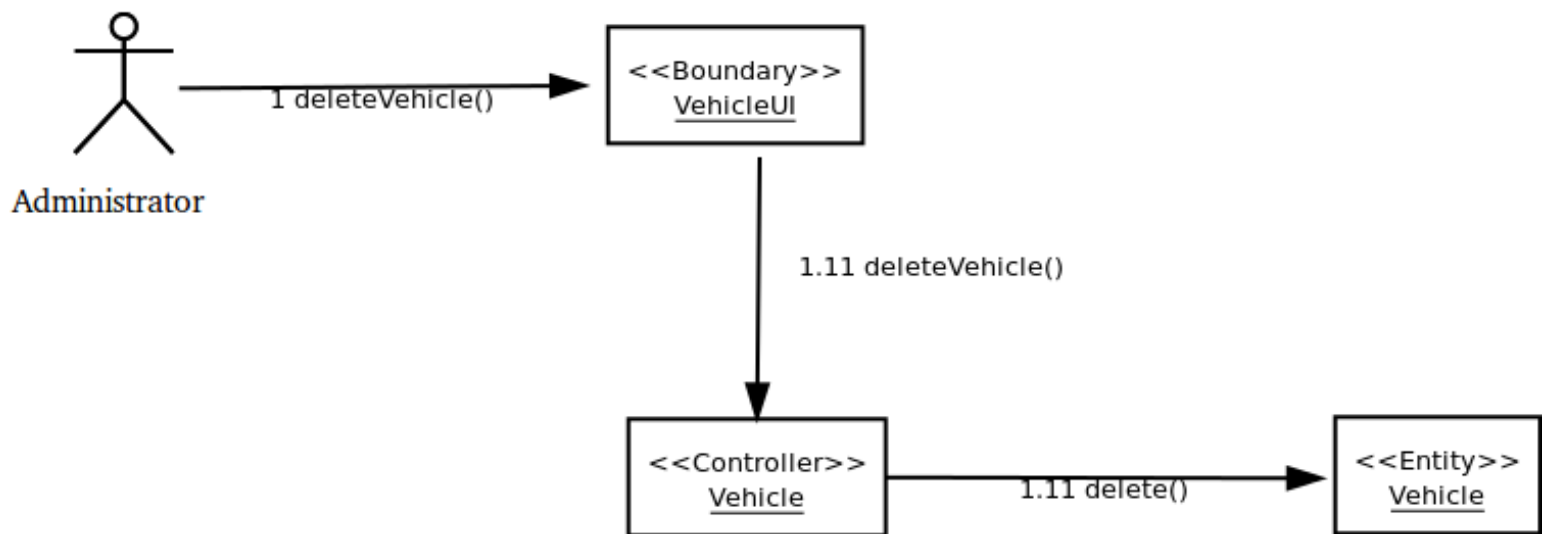
2.1.2.6 DeleteVehicle – Sequence Diagram

DeleteVehicle: Sequence Diagram



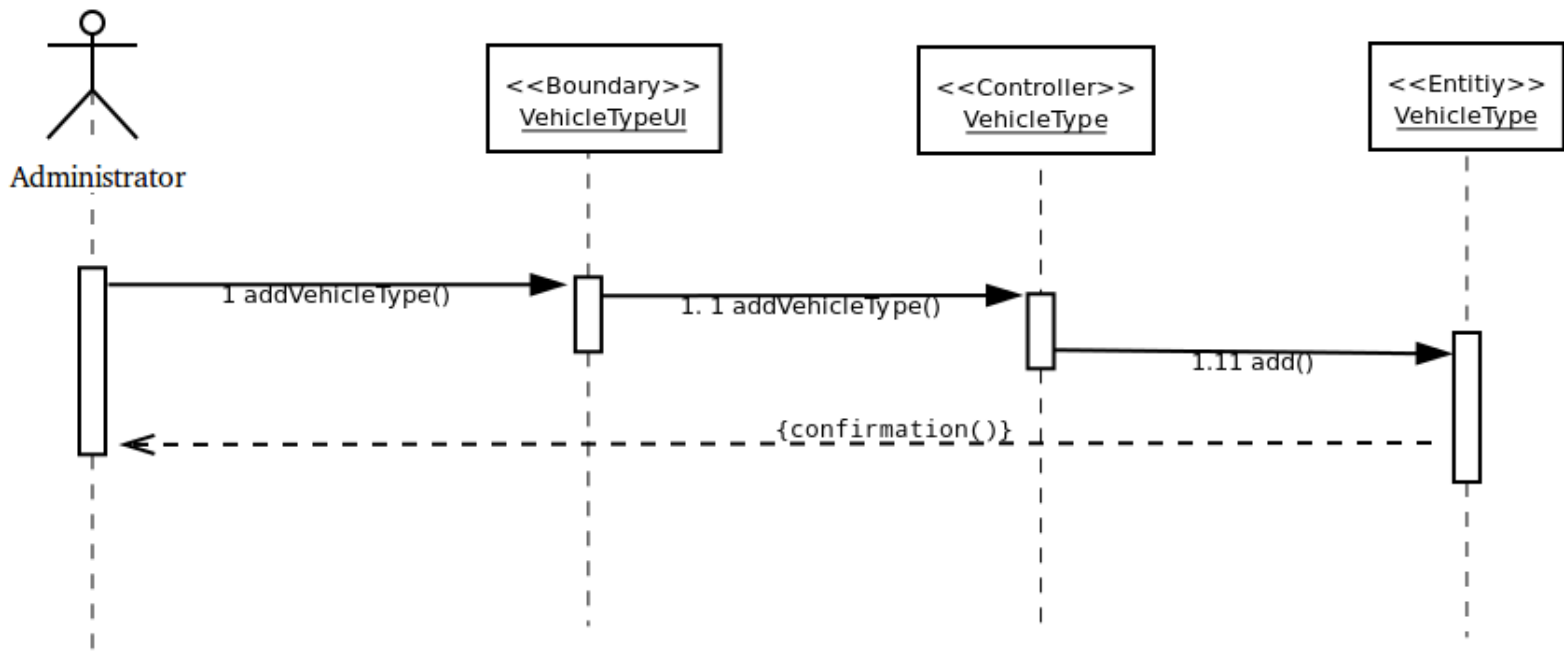
2.1.2.6 DeleteVehicle – Communication Diagram

DeleteVehicle: Collaboration Diagram



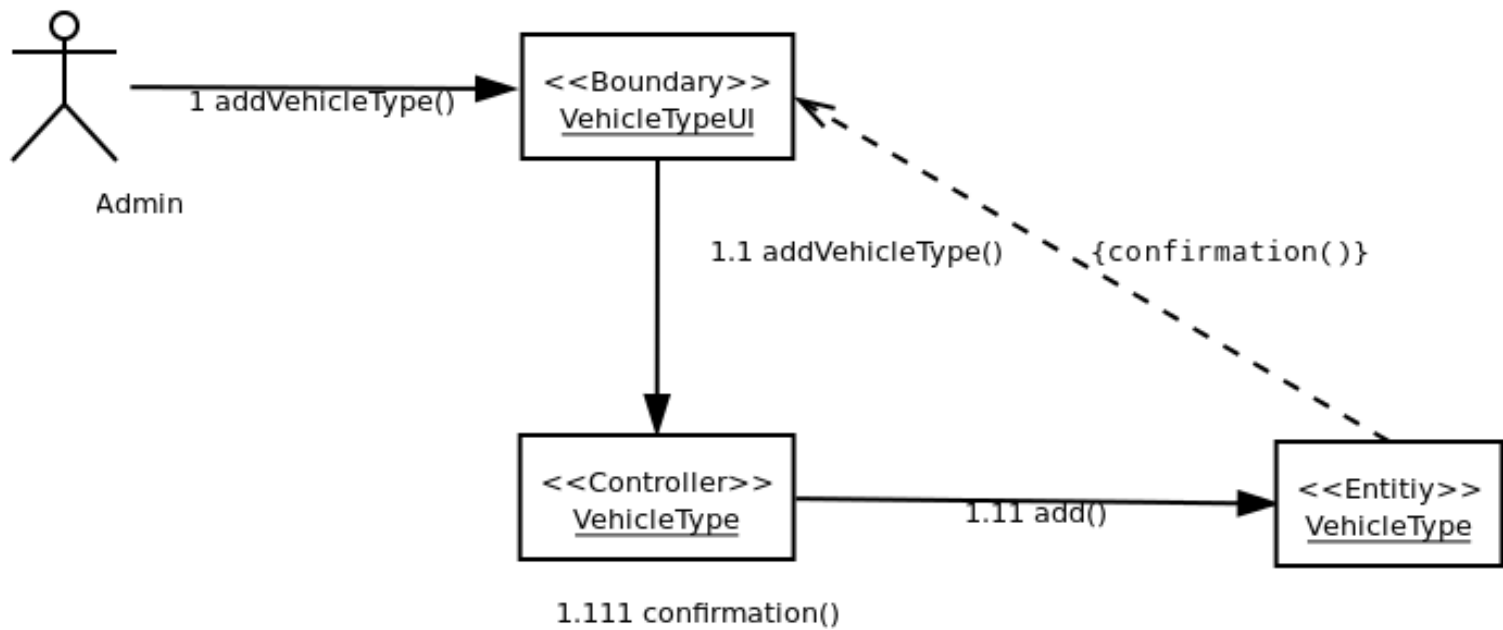
2.1.2.7 AddVehicleType – Sequence Diagram

AddVehicleType: Sequence Diagram



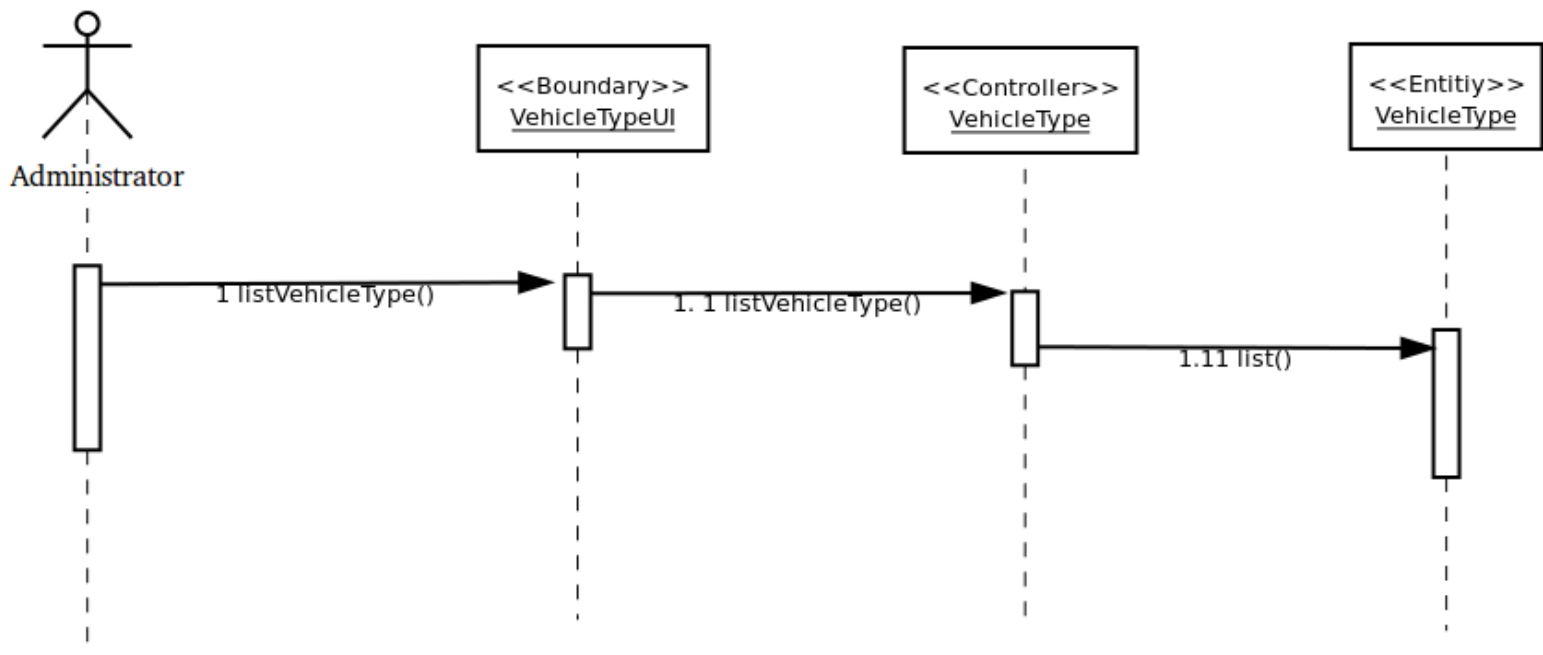
2.1.2.7 AddVehicleType – Communication Diagram

AddVehicleType: Collaboration Diagram



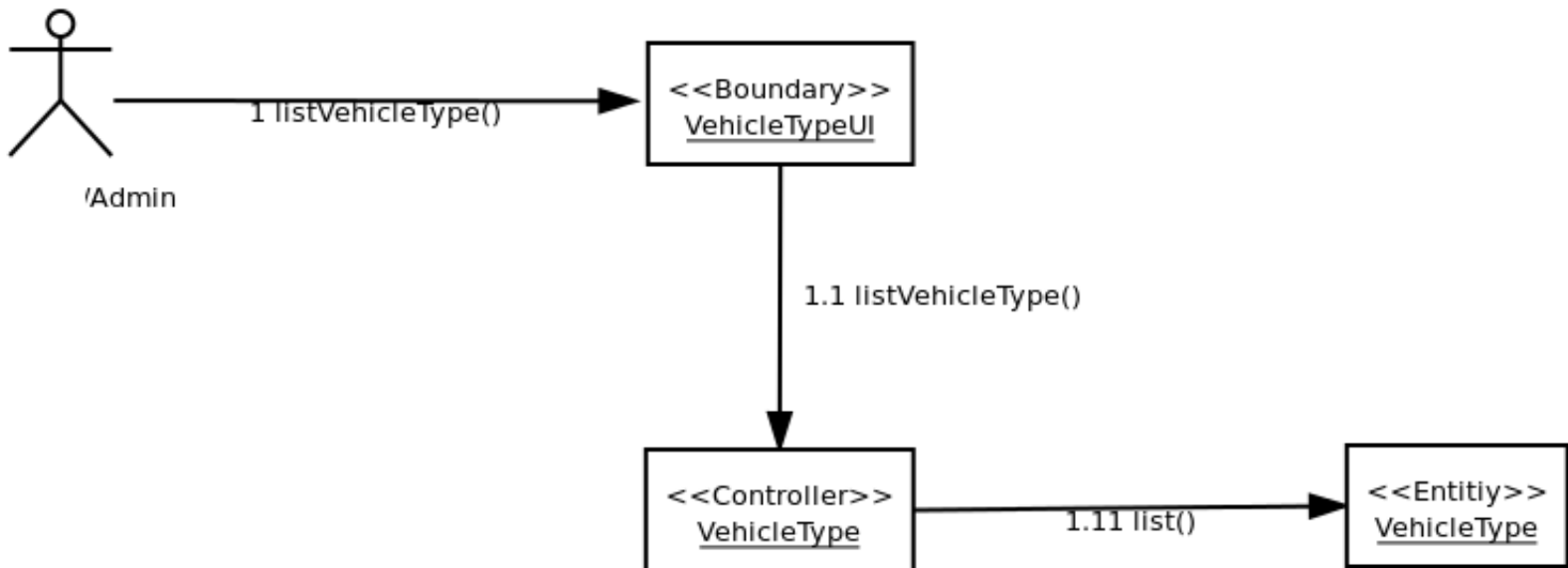
2.1.2.8 ListVehicleType – Sequence Diagram

ListVehicleType: Sequence Diagram



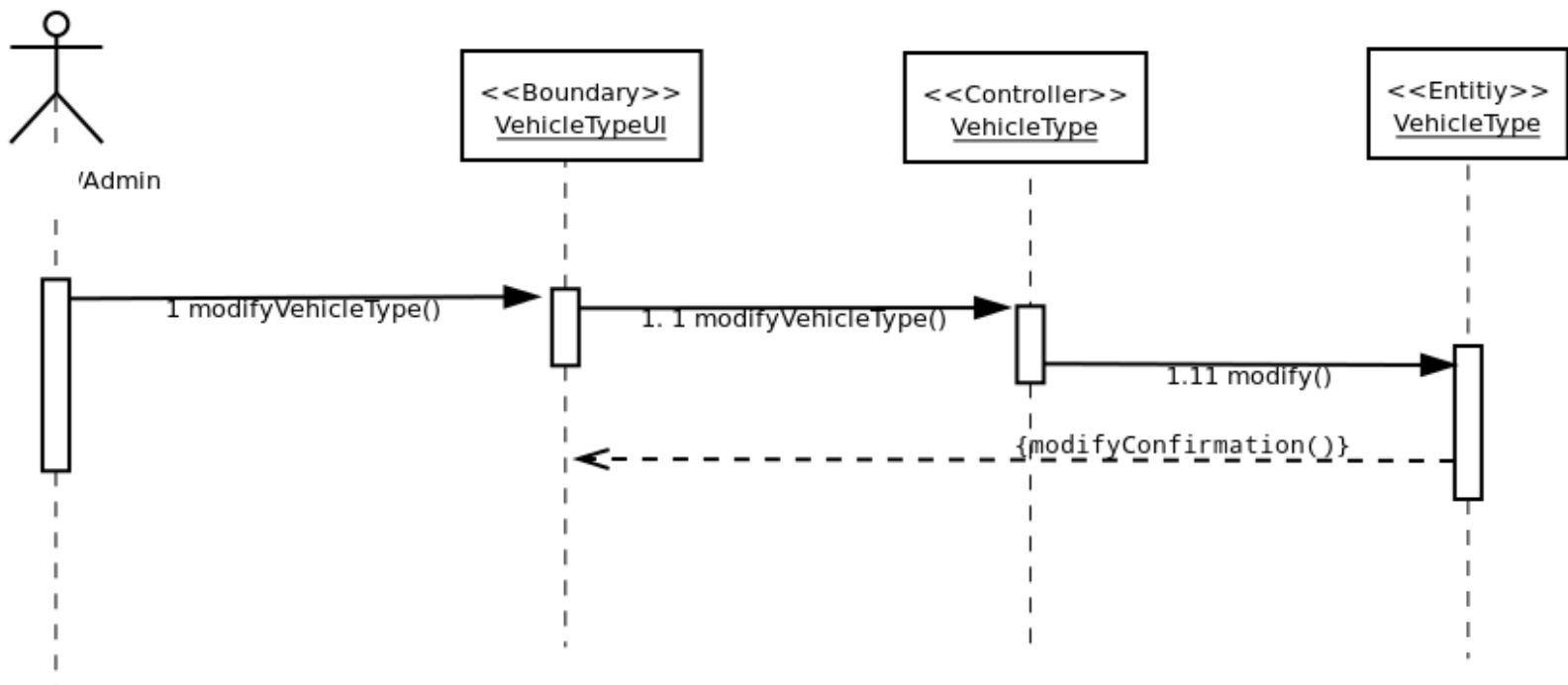
2.1.2.8 ListVehicleType – Communication Diagram

ListVehicleType: Collaboration Diagram

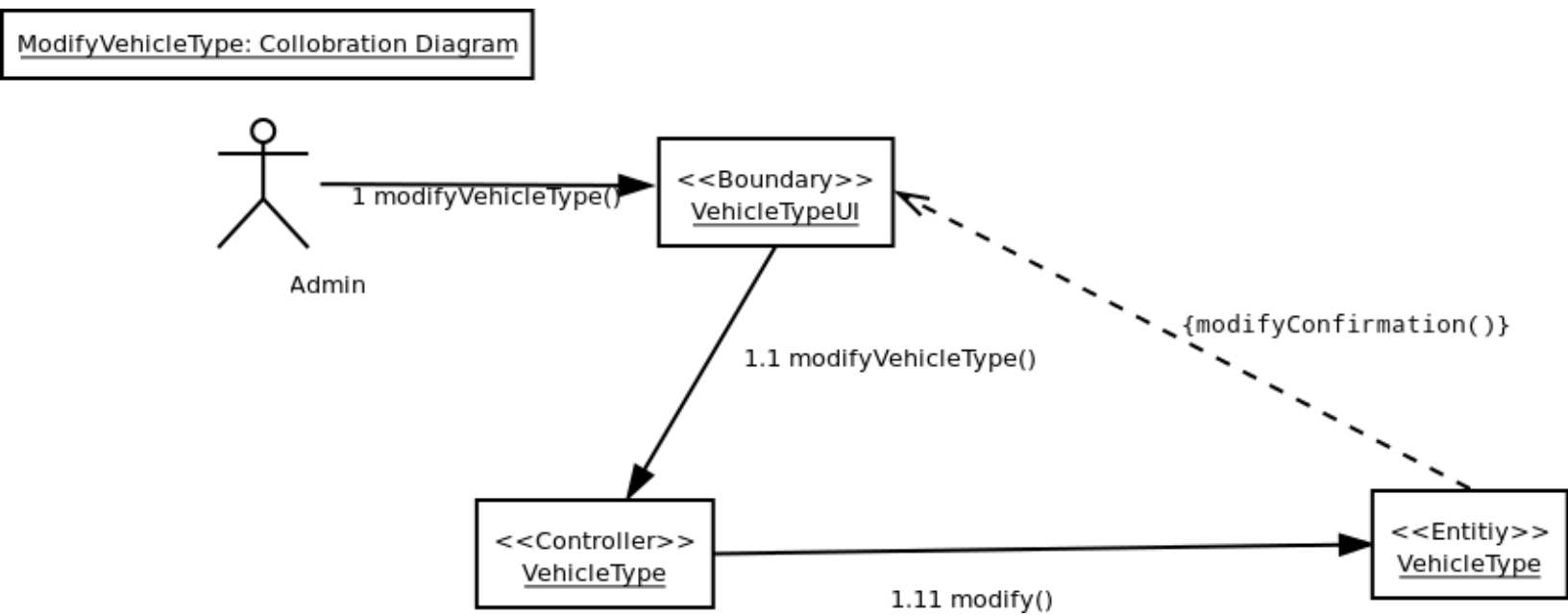


2.1.2.9 ModifyVehicleType – Sequence Diagram

ModifyVehicleType: Sequence Diagram

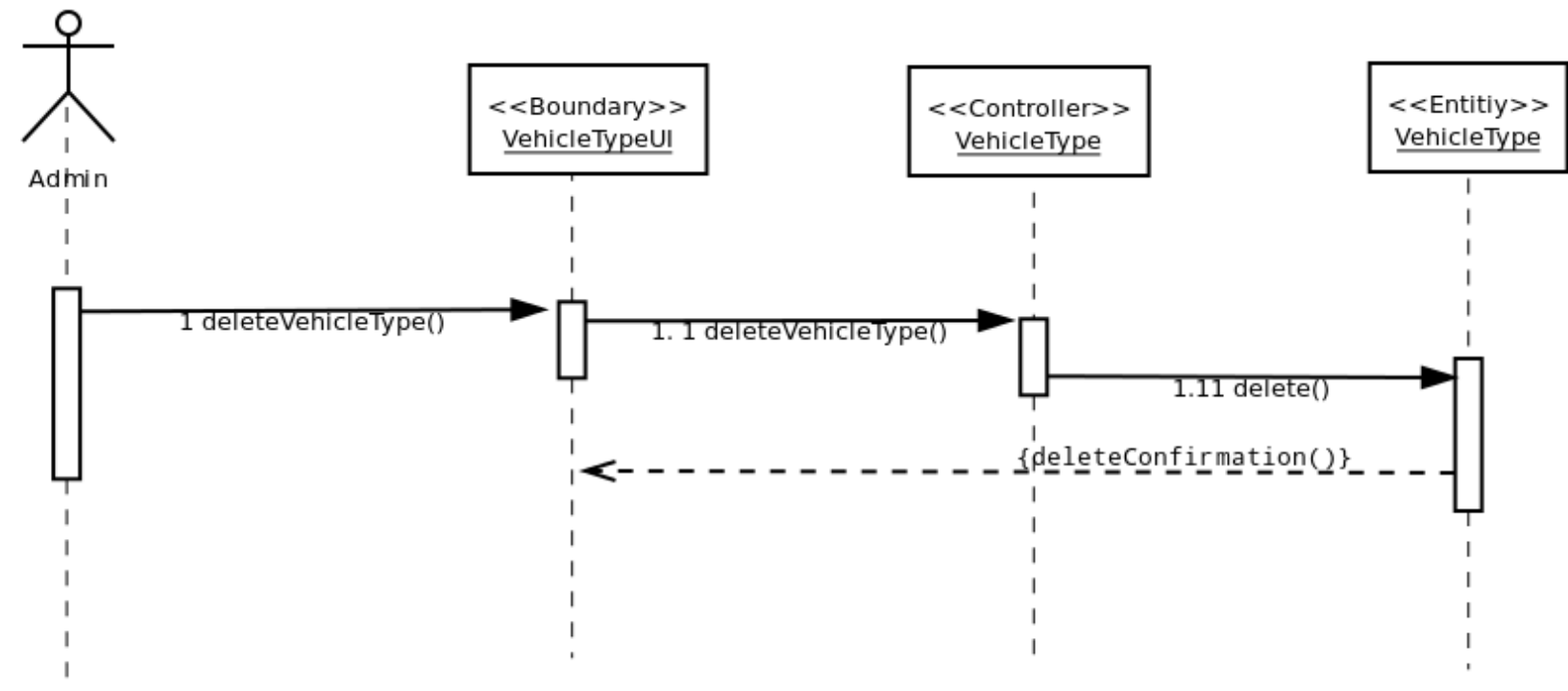


2.1.2.9 ModifyVehicleType – Communication Diagram

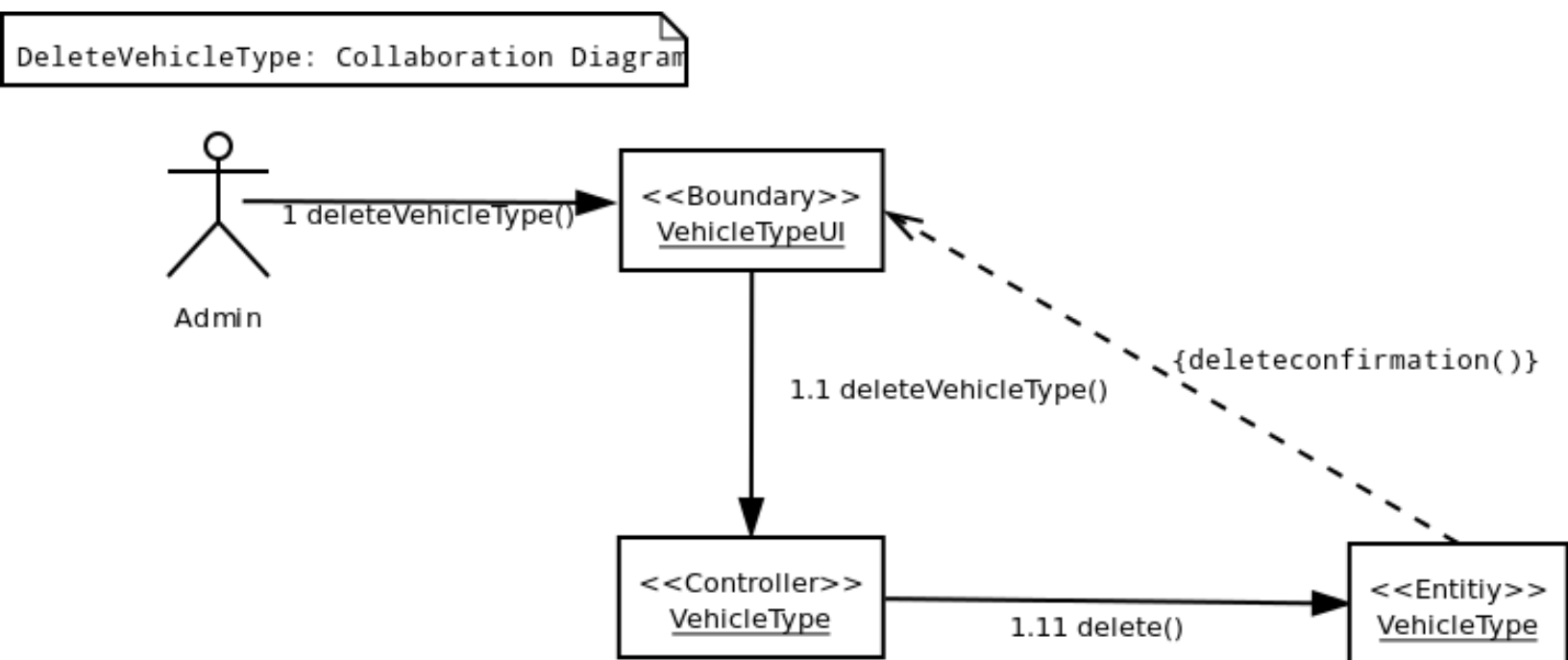


2.1.2.10 DeleteVehicleType – Sequence Diagram

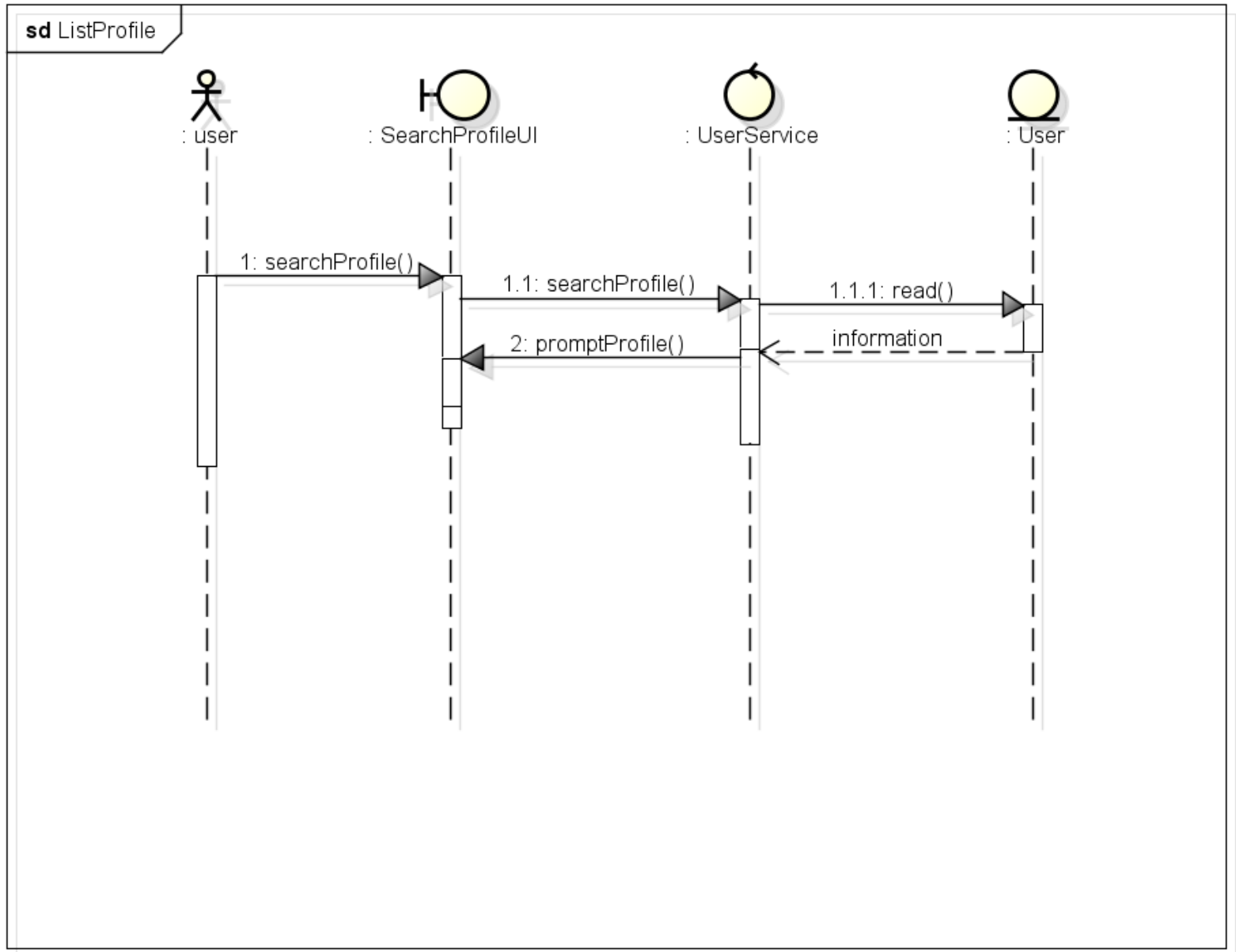
DeleteVehicleType: Sequence Diagram



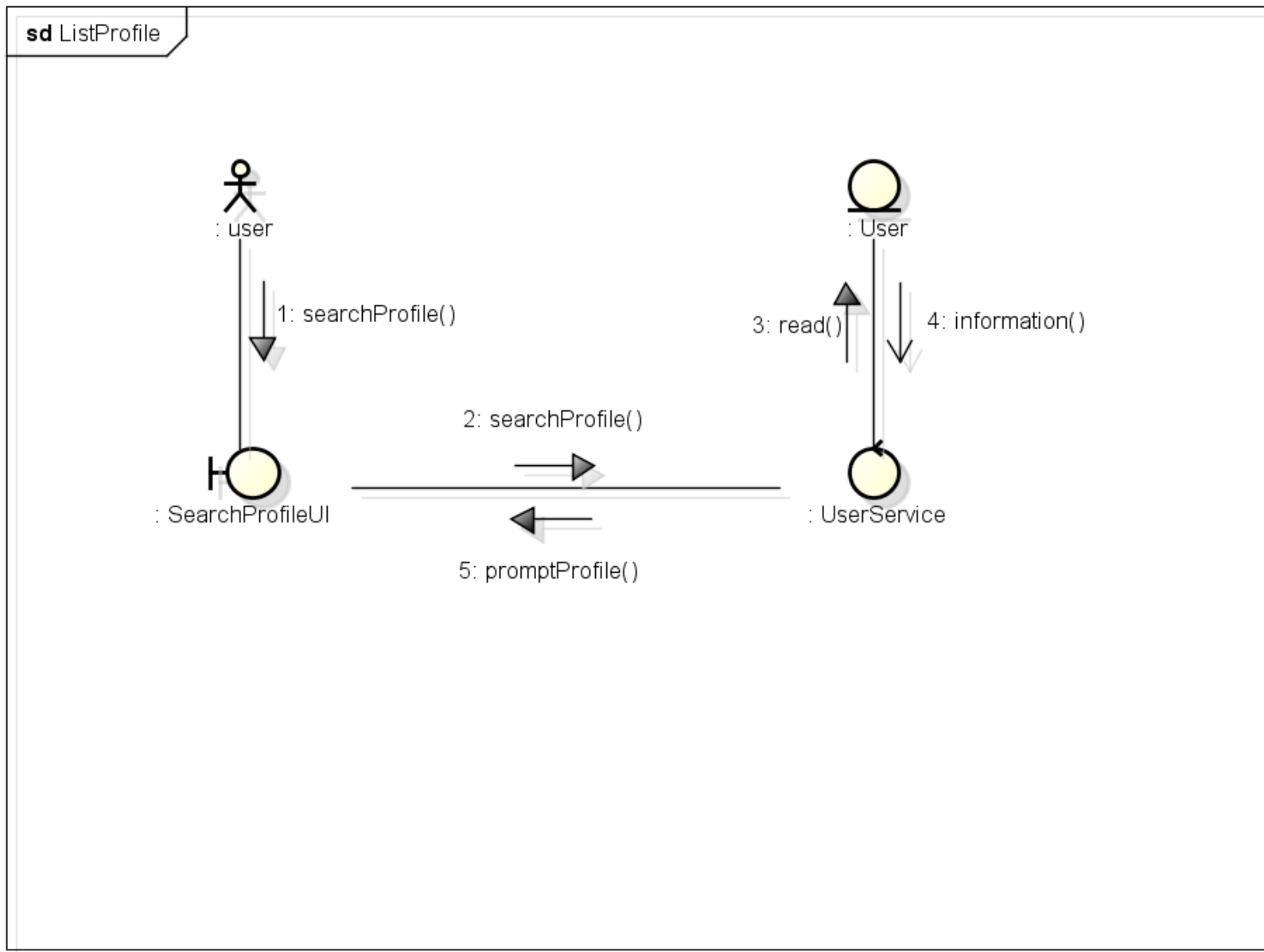
2.1.2.10 DeleteVehicleType – Communication Diagram



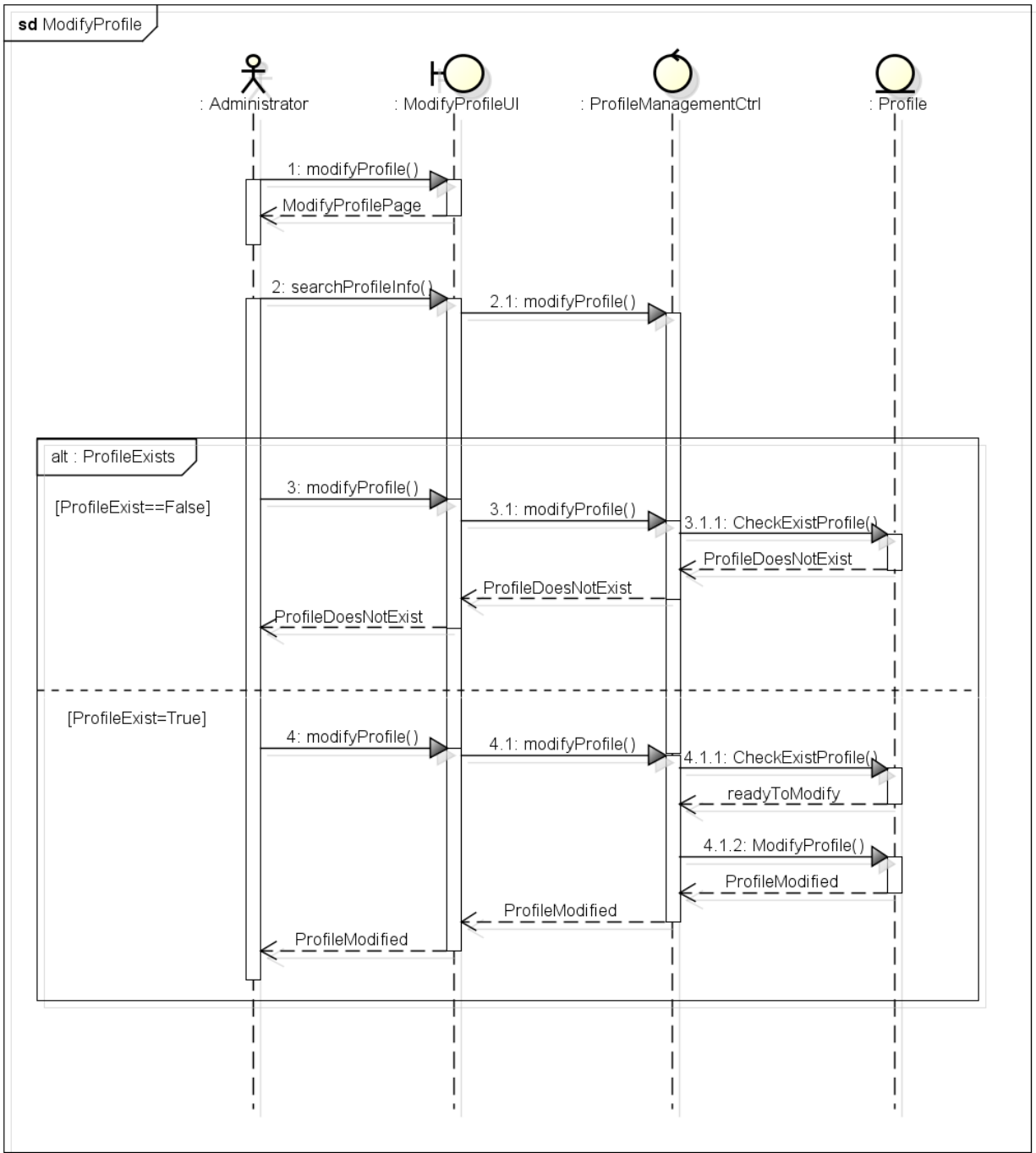
2.1.2.11 ListProfile – Sequence Diagram



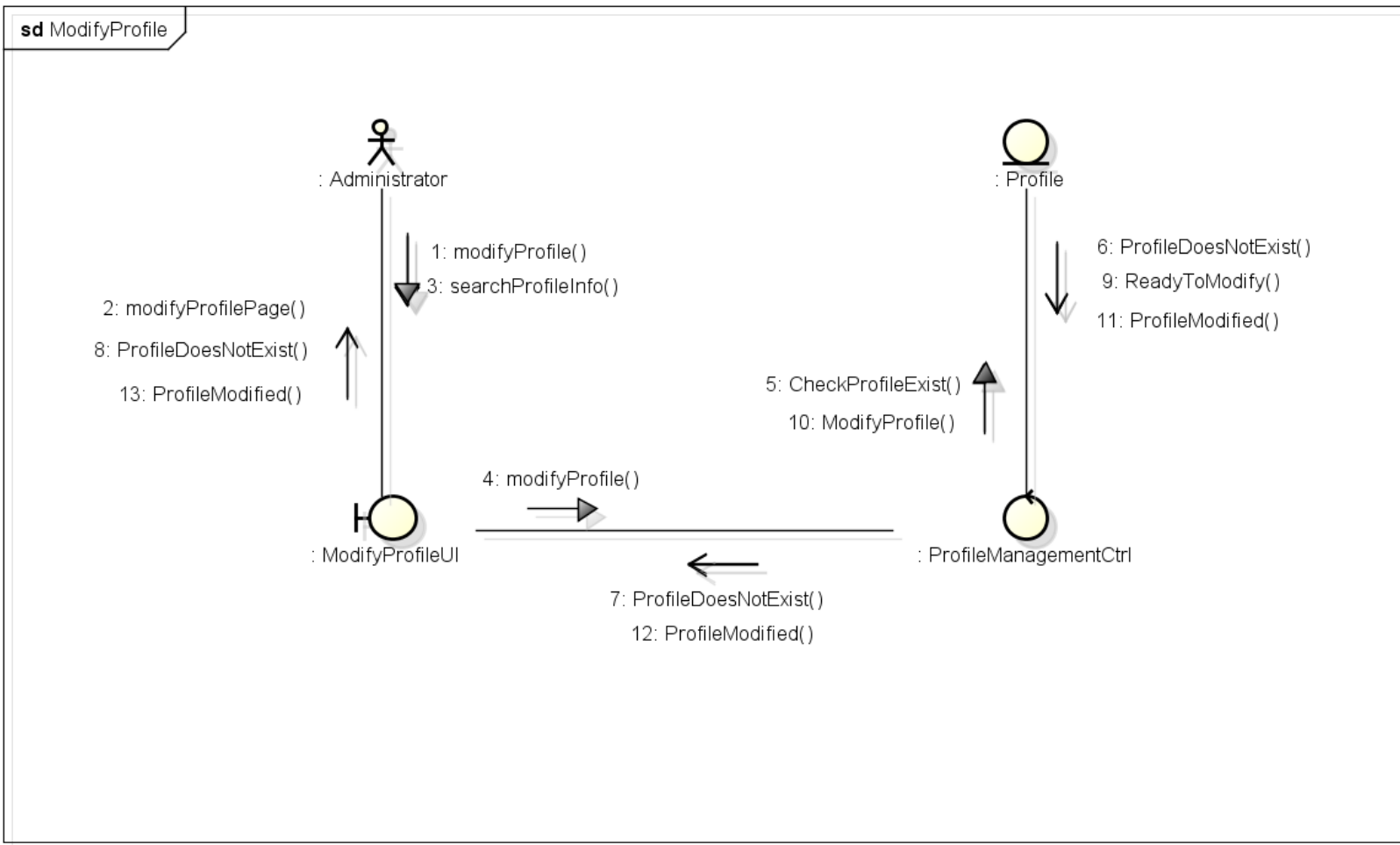
2.1.2.11 ListProfile – Communication Diagram



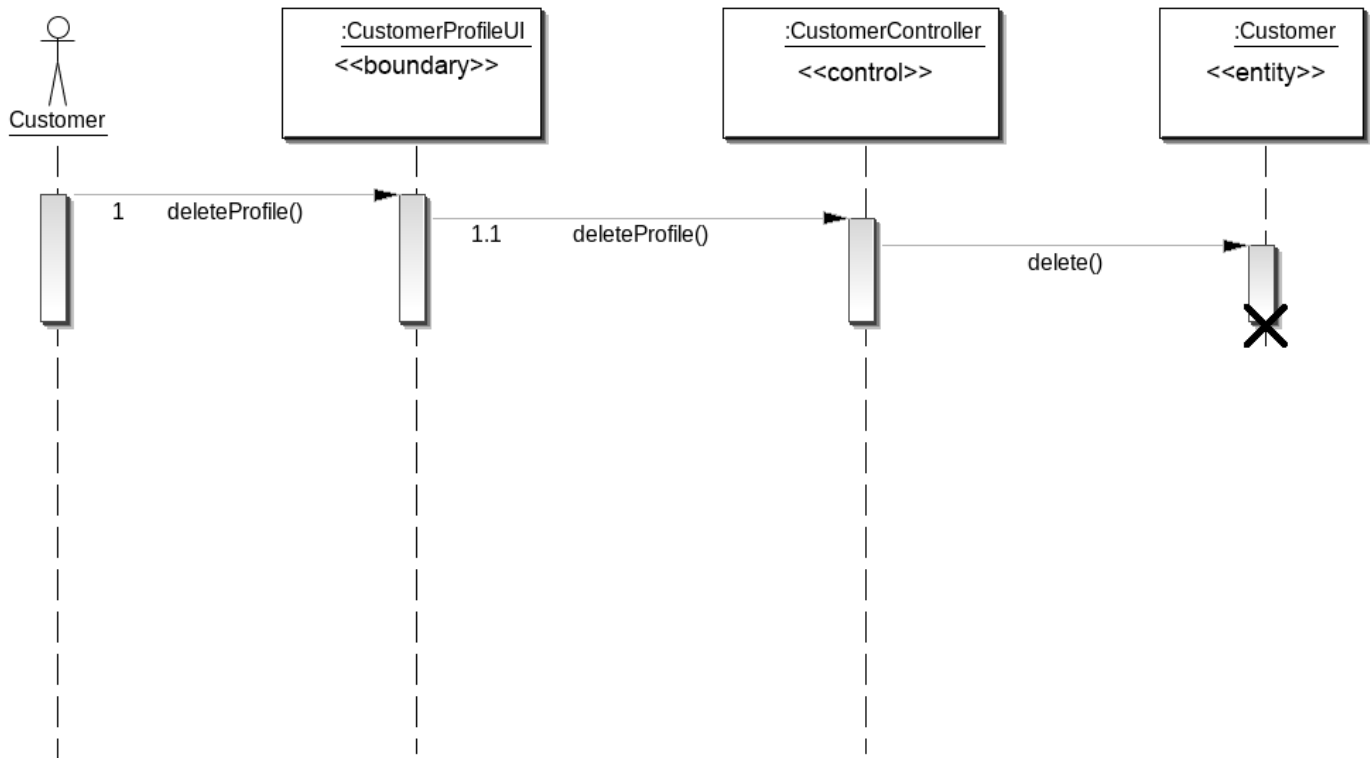
2.1.2.12 ModifyProfile – Sequence Diagram



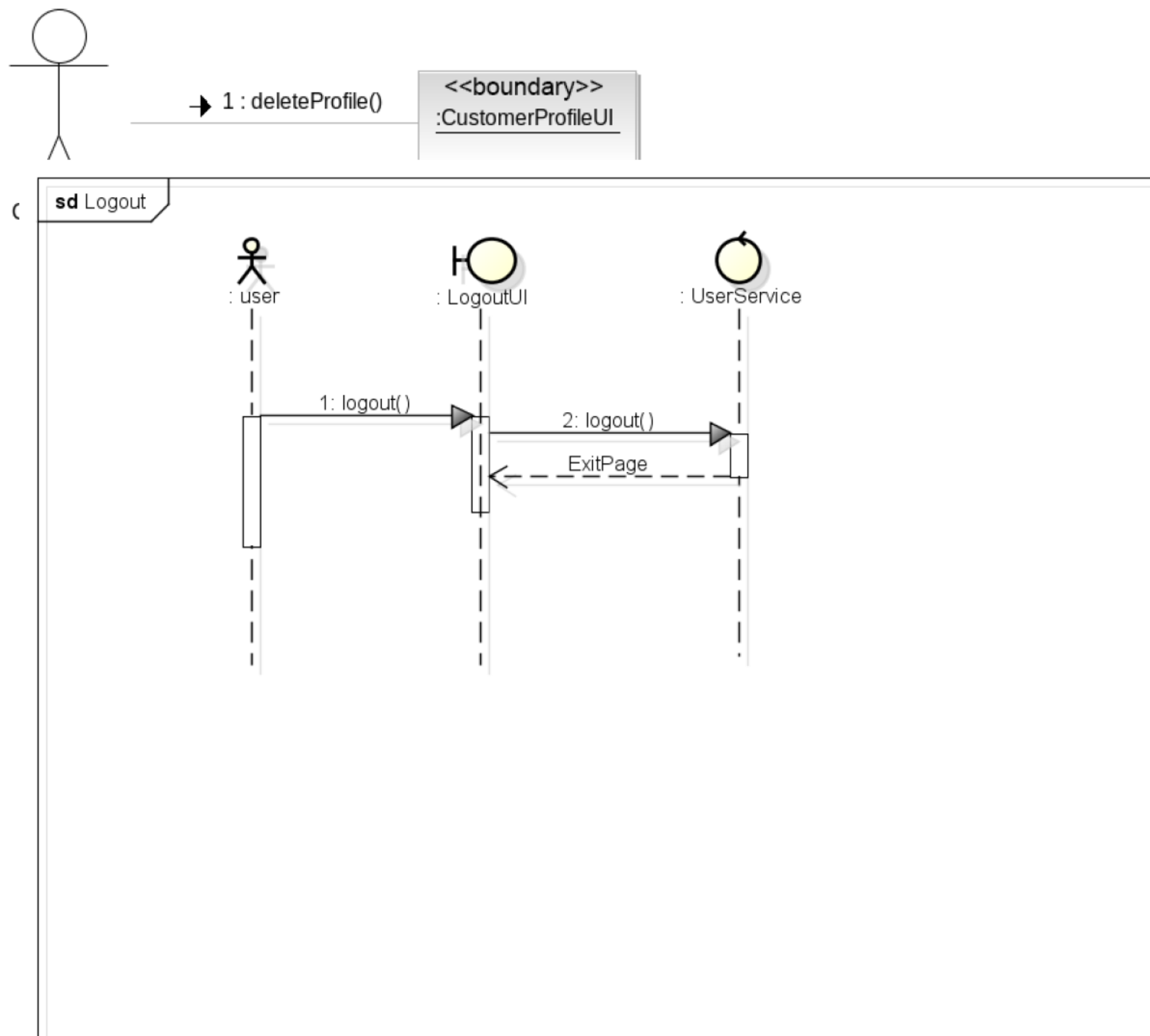
2.1.2.12 ModifyProfile – Communication Diagram



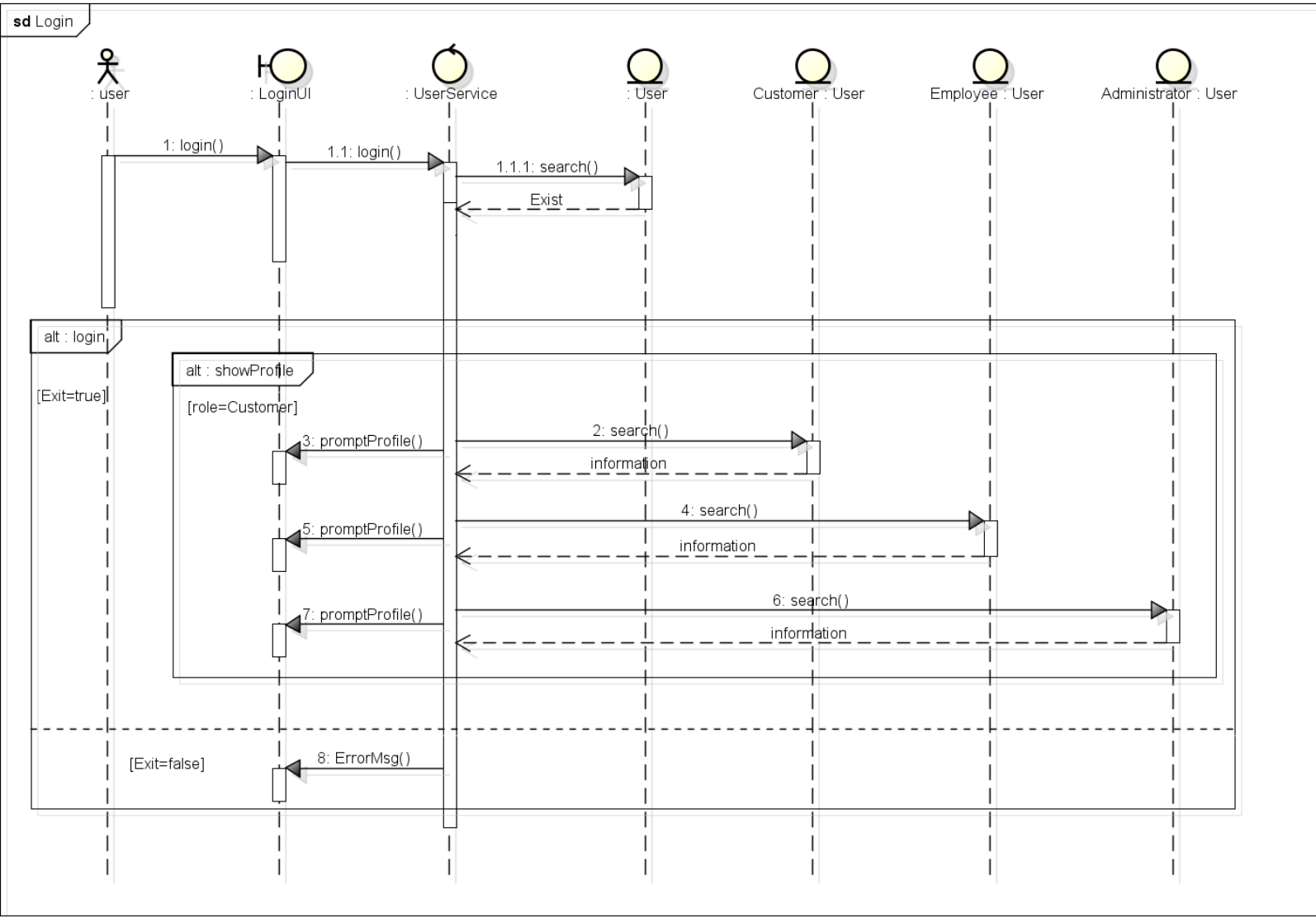
2.1.2.13 DeleteProfile – Sequence Diagram



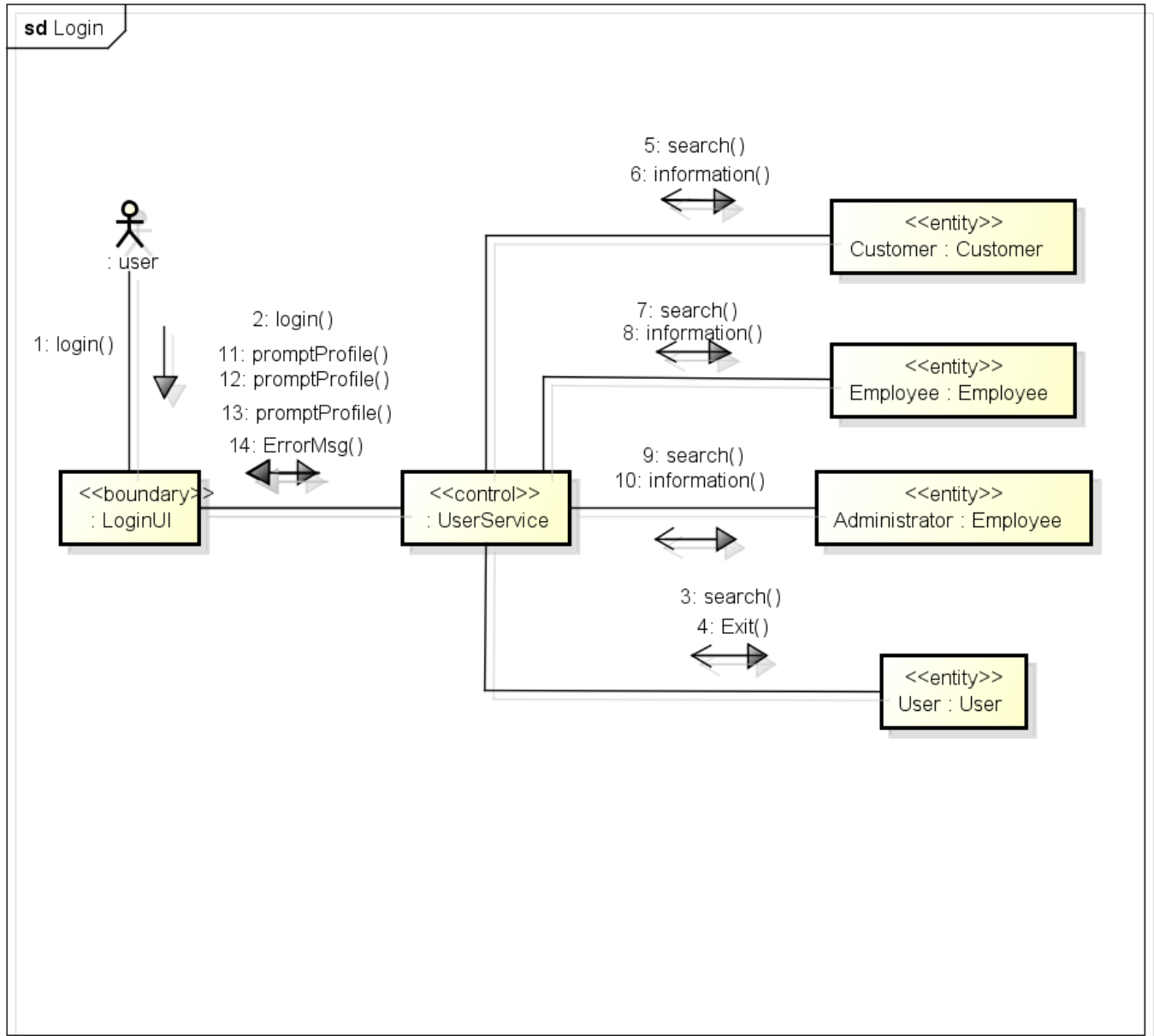
2.1.2.13 DeleteProfile – Communication Diagram



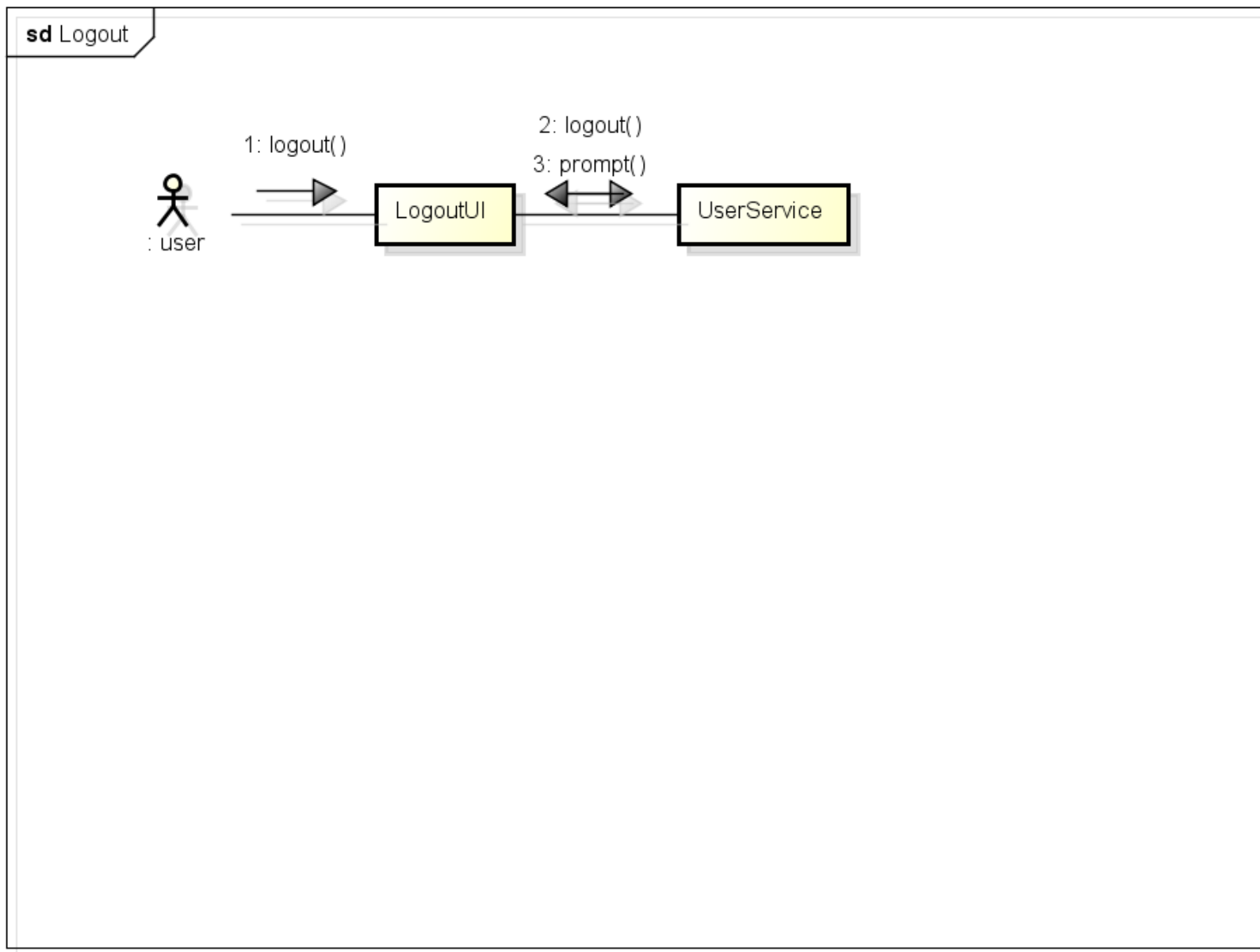
2.1.2.14 Login – Sequence Diagram



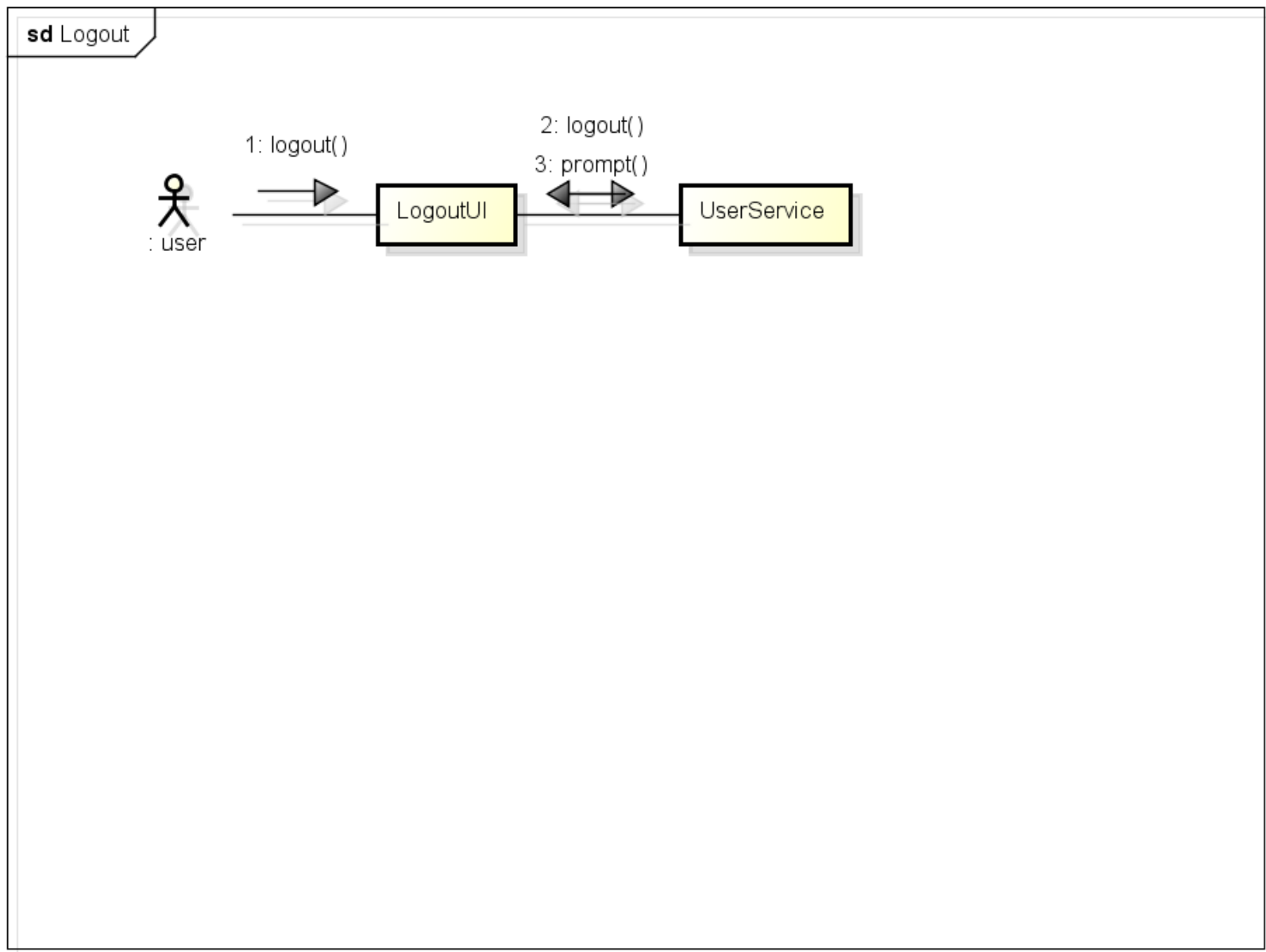
2.1.2.14 Login – Communication Diagram



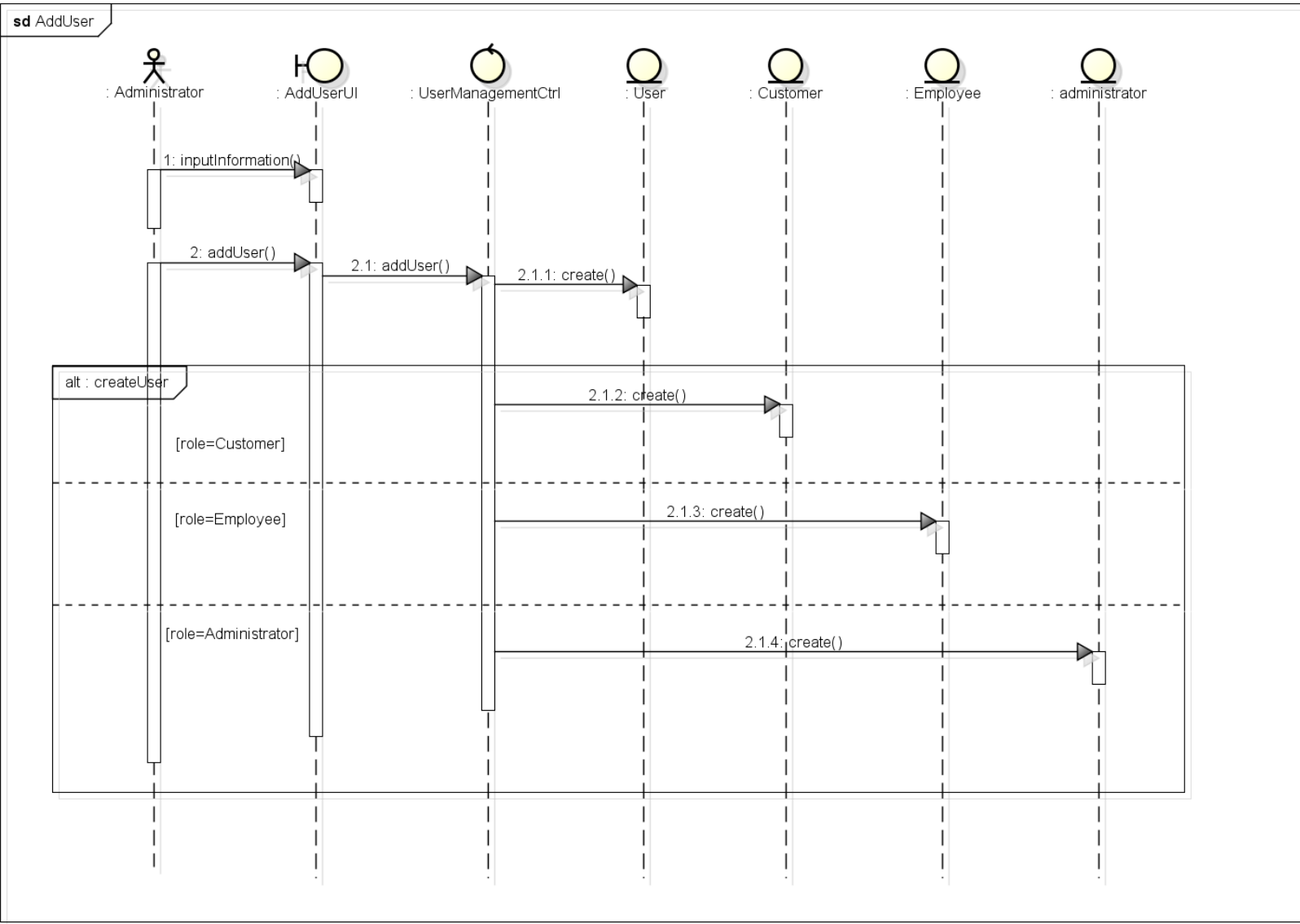
2.1.2.15 Logout – Sequence Diagram



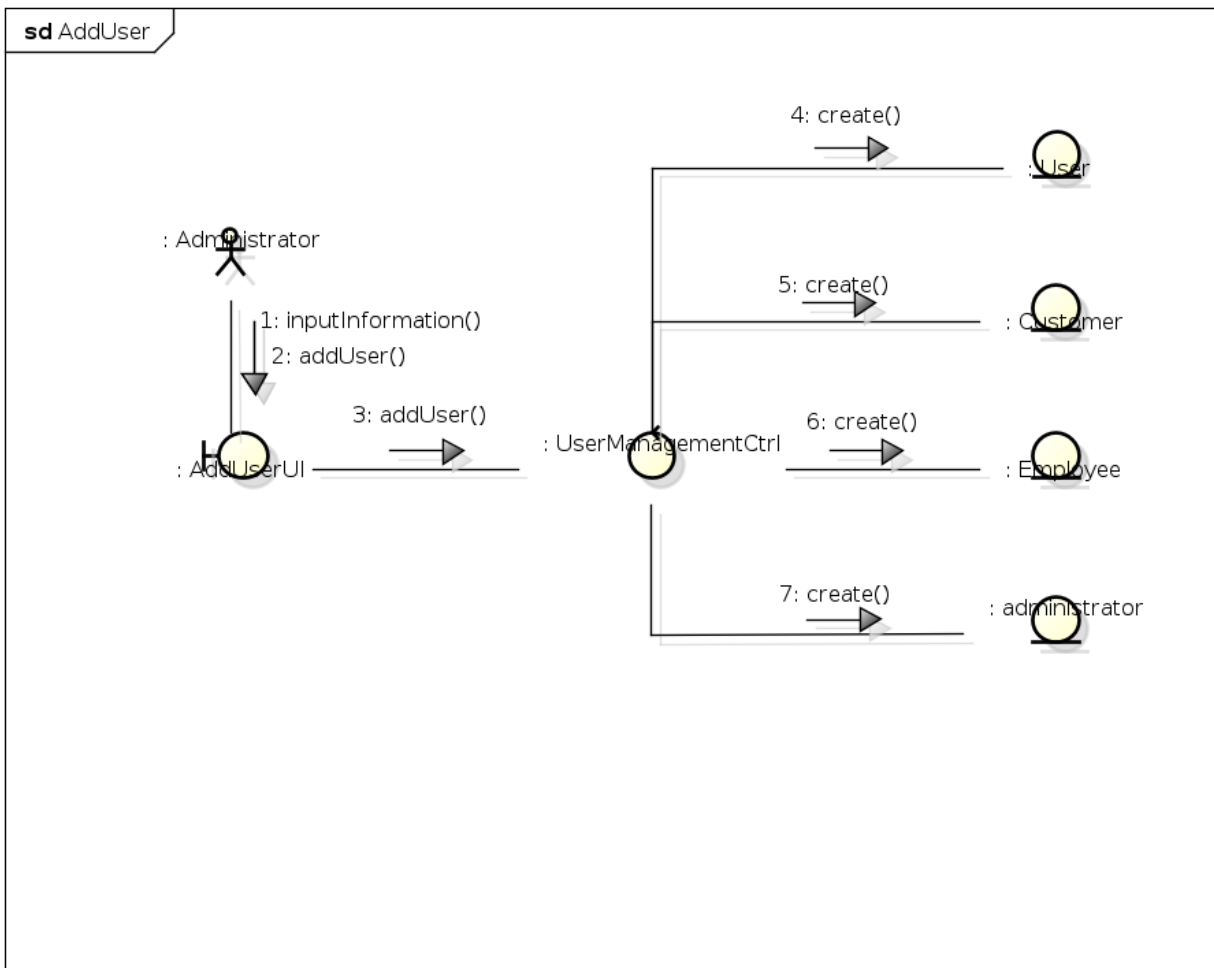
2.1.2.15 Logout – Communication Diagram



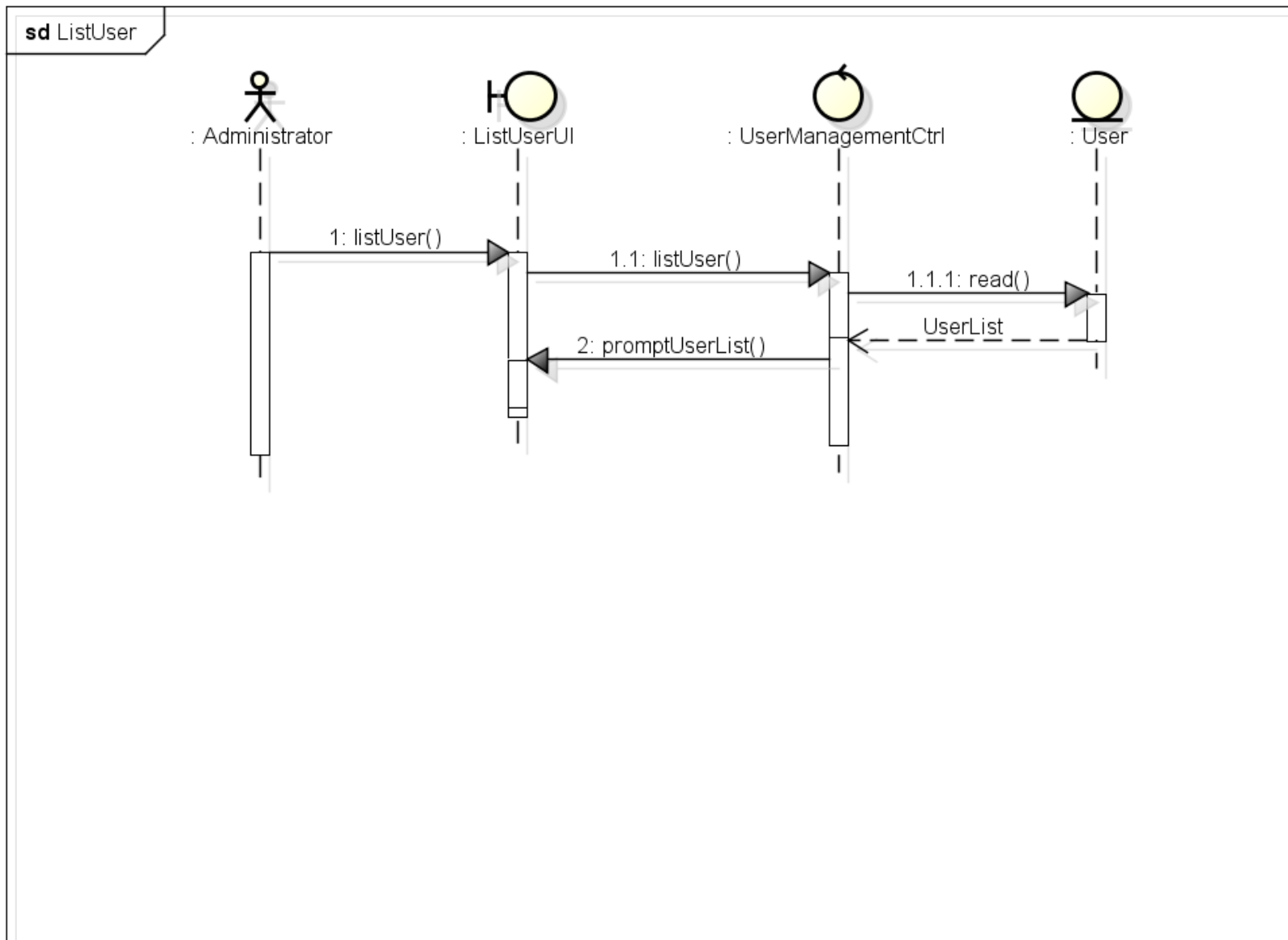
2.1.2.16 AddUser – Sequence Diagram



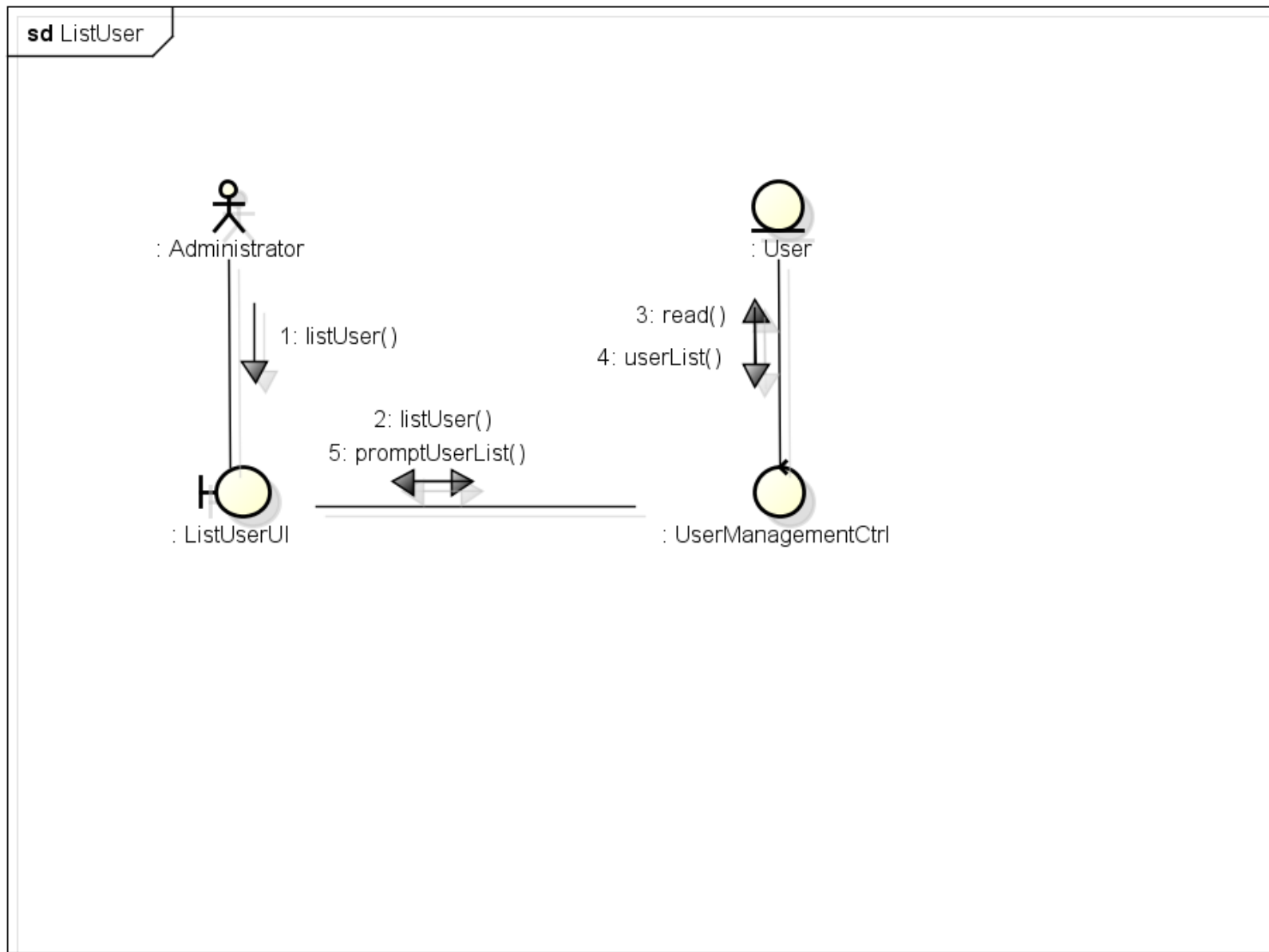
2.1.2.16 AddUser – Communication Diagram



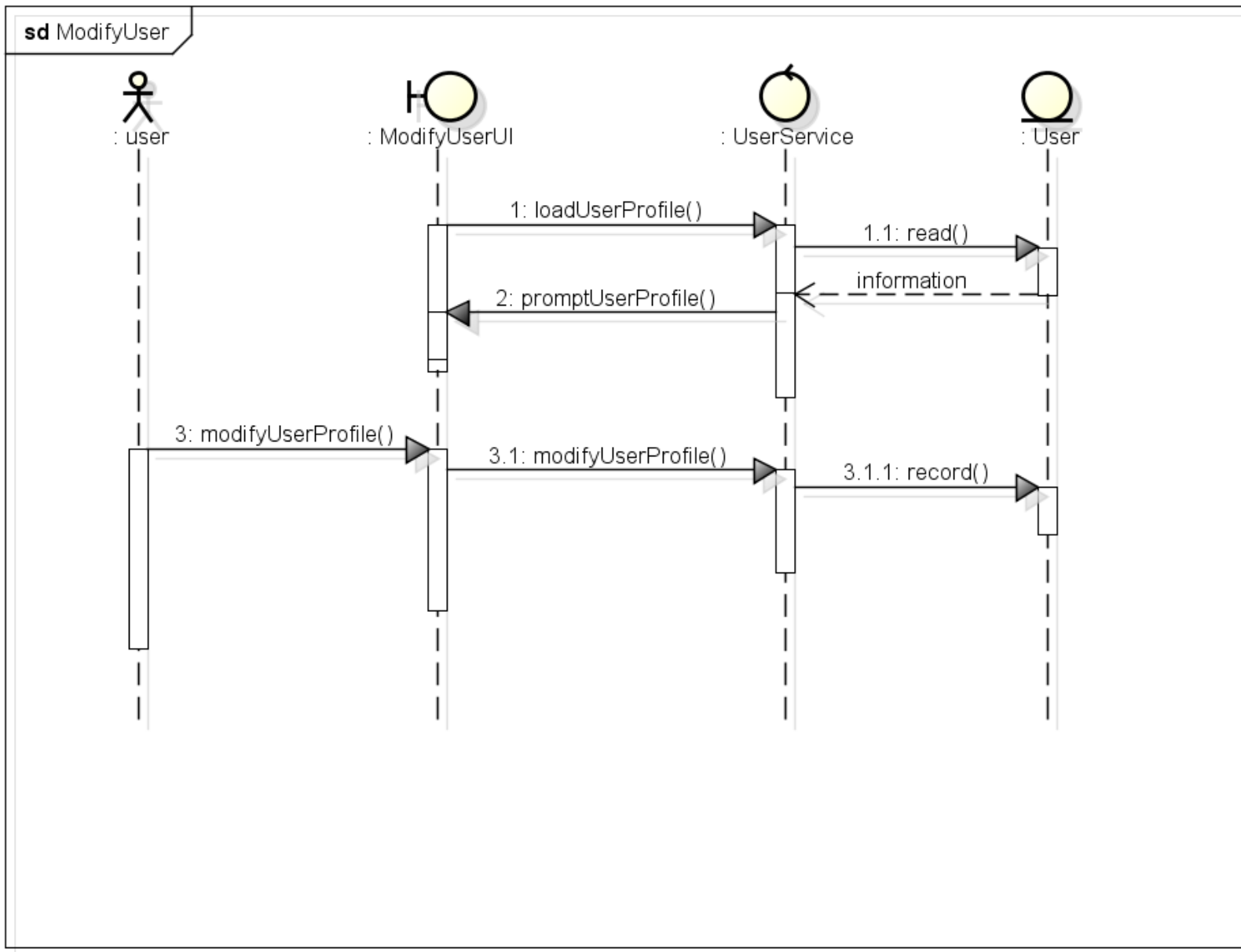
2.1.2.17 ListUser – Sequence Diagram



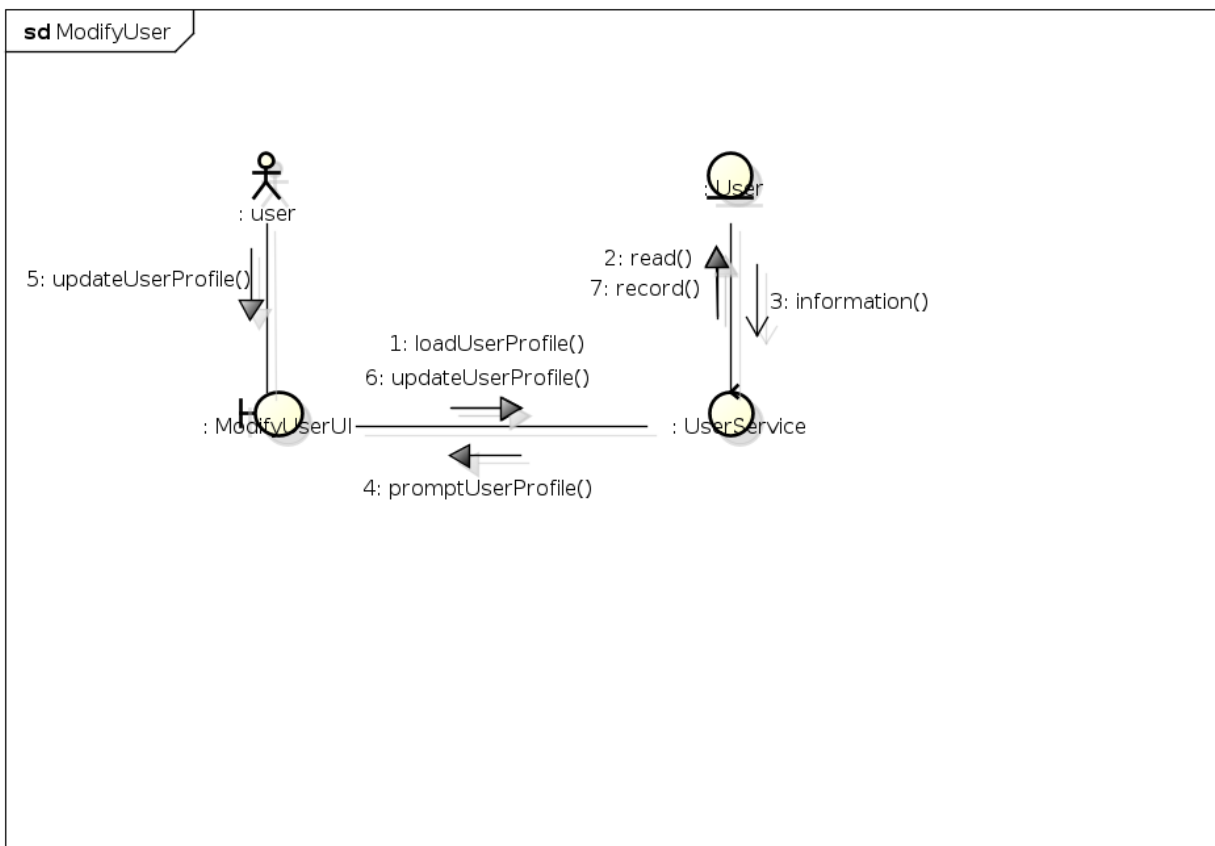
2.1.2.17 ListUser – Communication Diagram



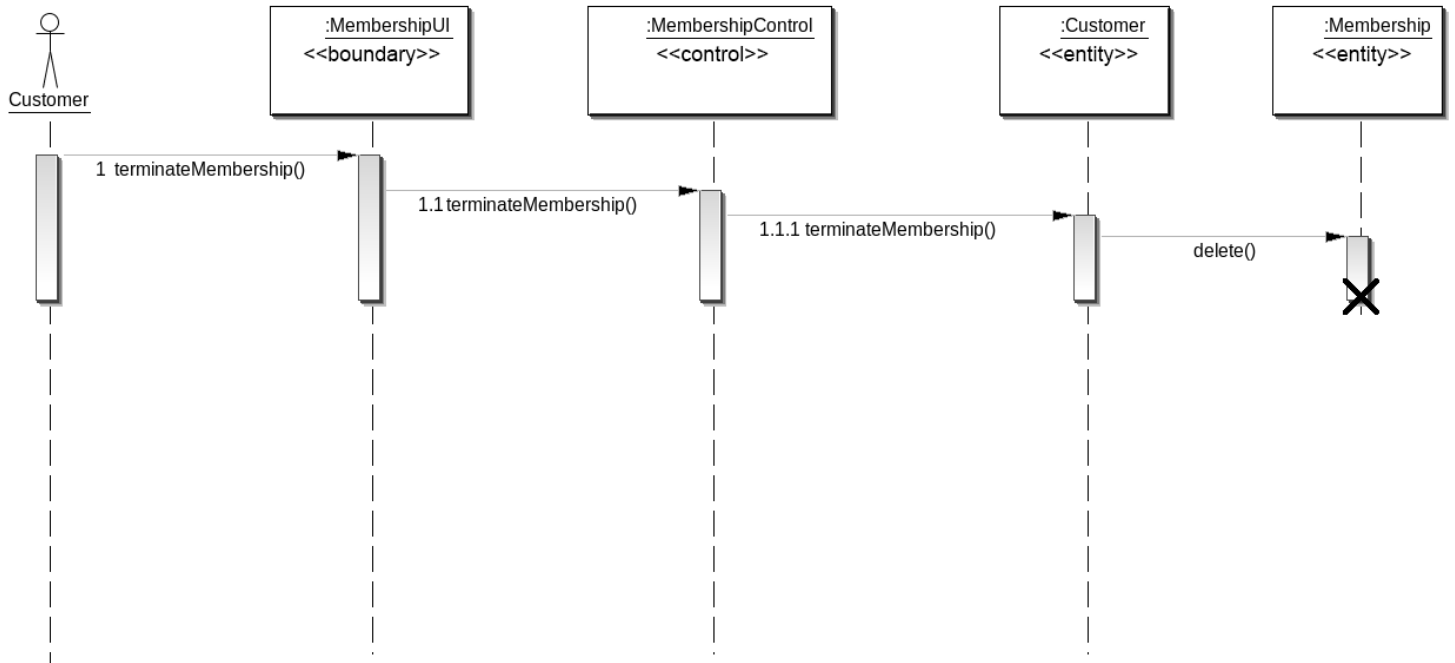
2.1.2.18 ModifyUser – Sequence Diagram



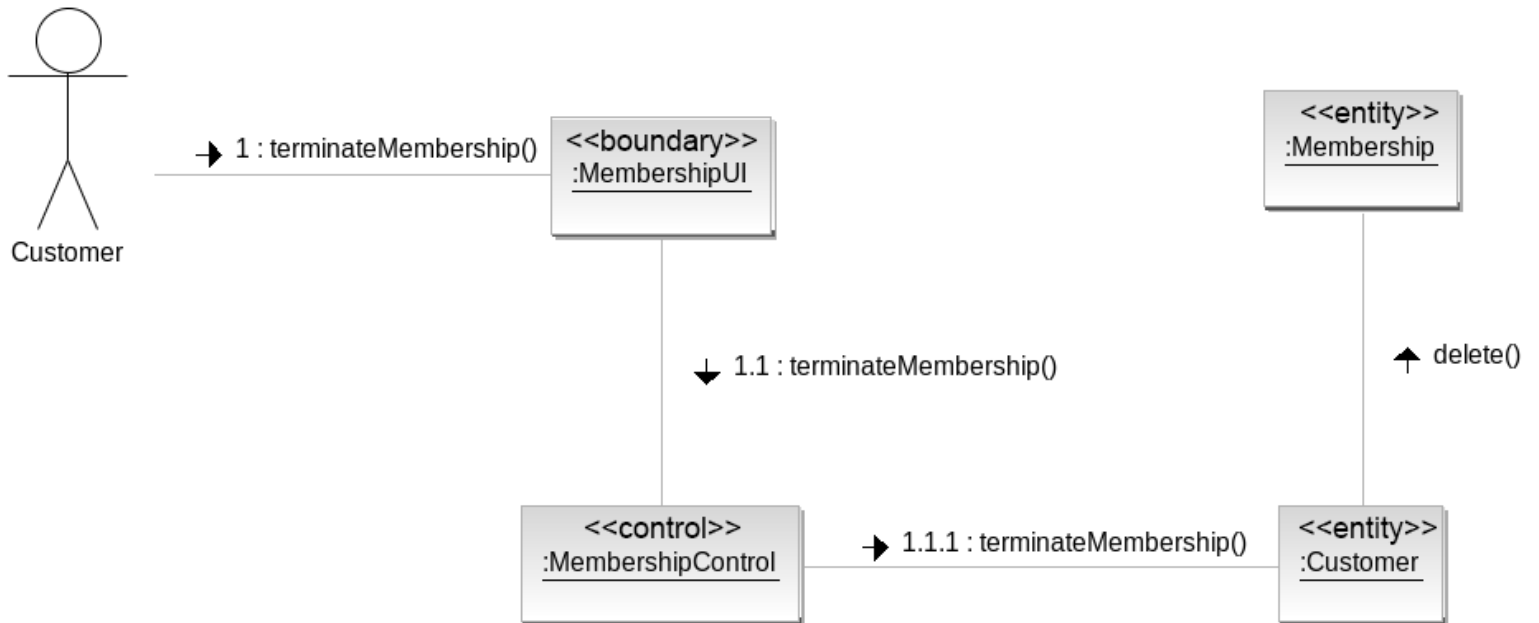
2.1.2.18 ModifyUser – Communication Diagram



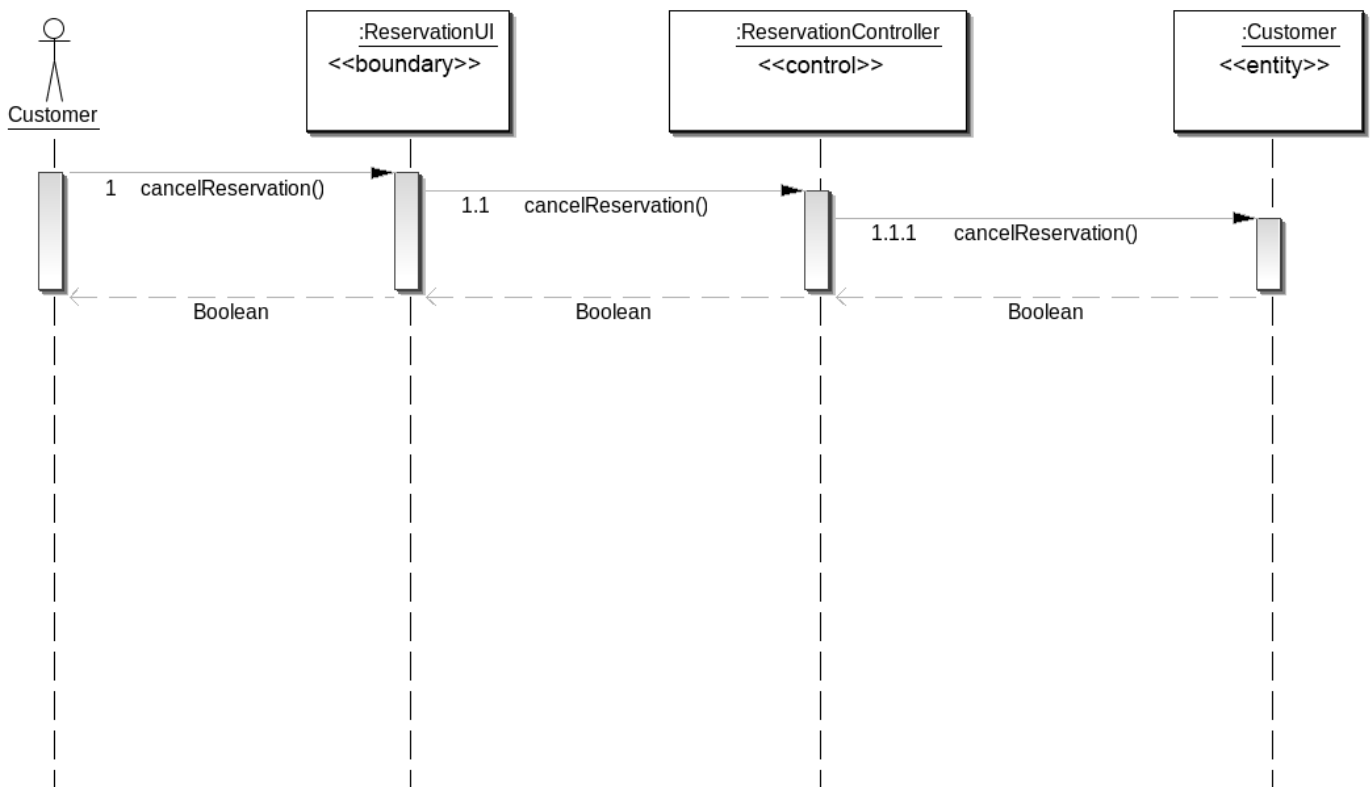
2.1.2.19 TerminateMembership – Sequence Diagram



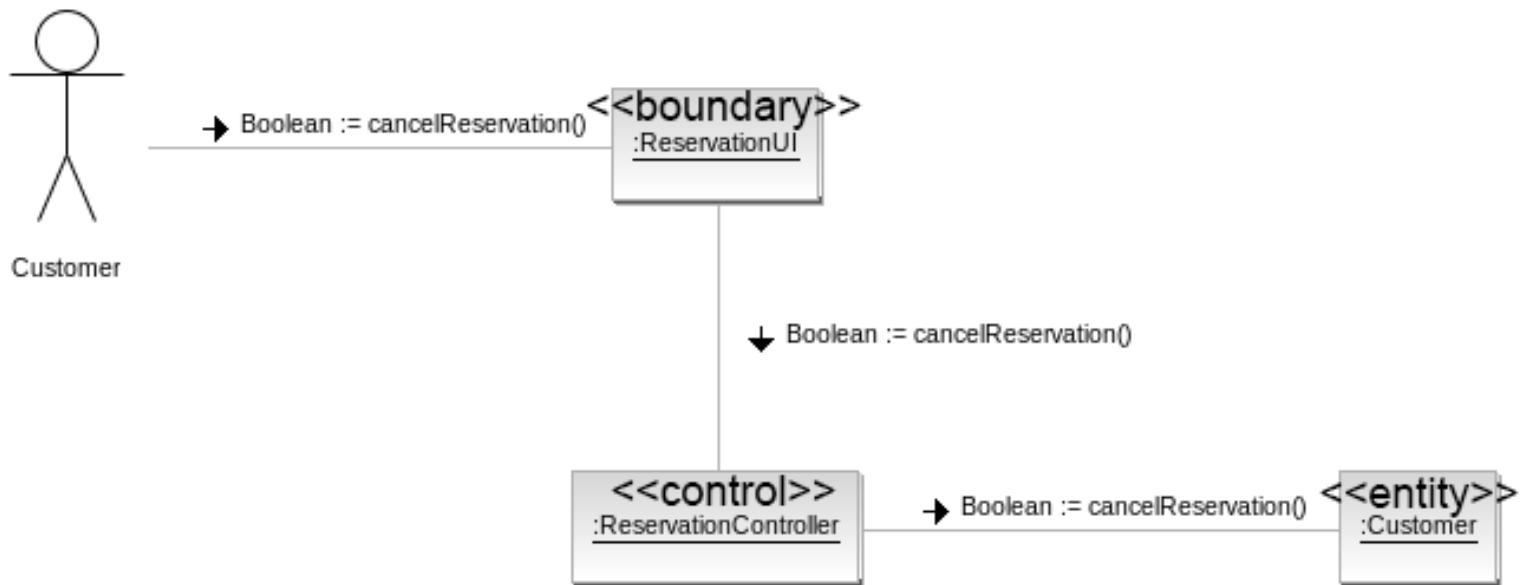
2.1.2.19 TerminateMembership – Communication Diagram



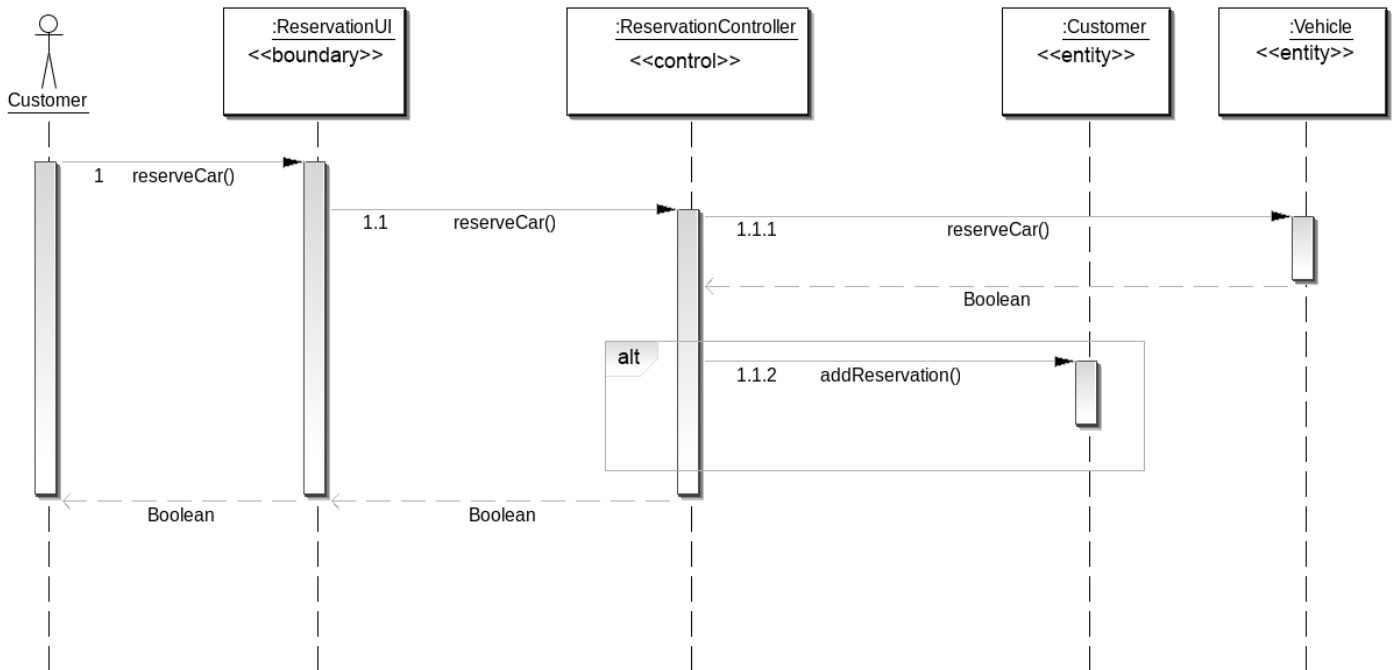
2.1.2.20 CancelReservation – Sequence Diagram



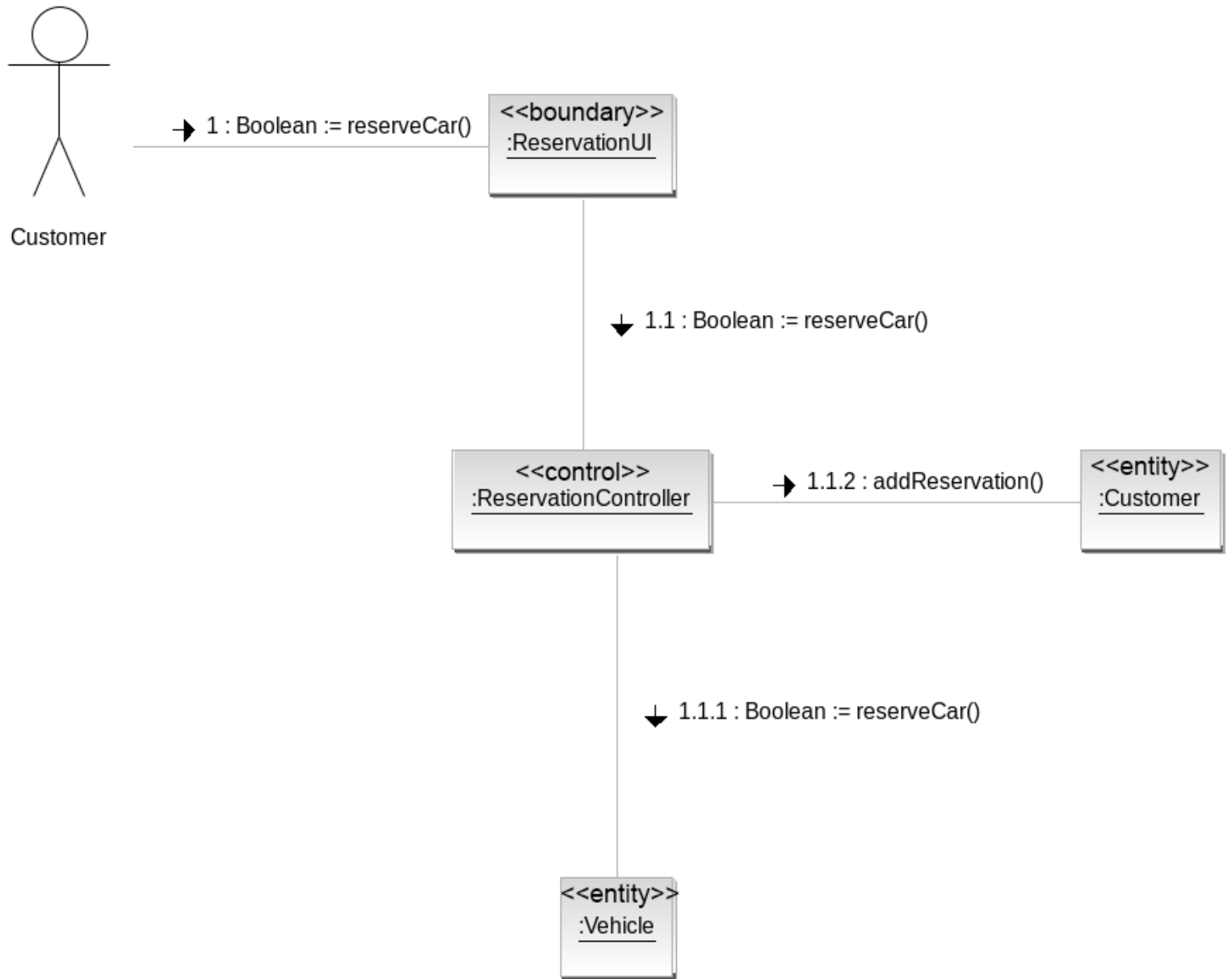
2.1.2.20 CancelReservation – Communication Diagram



2.1.2.21 ReserveCar – Sequence Diagram



2.1.2.21 ReserveCar – Communication Diagram



3. Glossary

Term	Definition
Entity Classes	Model the information handled by the Car rental system. (Ex: Vehicles, Customer Profiles, etc.)
Boundary Classes	Handle communication between actors and system internals. (Ex: Administrator-Database Communication is an example)
Control Classes	Classes that handle the flow of control for our use cases. (Ex: Switching UI display from Search to Search Results)
Control Service Package	A bundle of Control Classes
Boundary Services Package	A bundle of Boundary Classes
Entity Service Package	A bundle of Entity Classes
{User Type}.boundary	Object model we have reference with the following set-up these are a reference to a particular user{administrator or customer} it allows us to specify which set of boundary classes we are addressing. (Ex: administrator .boundary package }
{UserType}.control	Object model we have reference with the following set-up these are a reference to a particular user{administrator or customer} it allows us to specify which set of boundary classes we are addressing. (Ex: administrator.control }
{UserType}.entity	Object model we have reference with the following set-up these are a reference to a particular user{administrator or customer} it allows us to specify which set of boundary classes we are addressing. (Ex: administrator.boundary)
{ActionWord}UI	The Following notation within our notation defines the actual action, and user interface displayed with the associated with our action. This gives our system readability and understanding (Ex: SearchProfileUI, VehicleTypeUI, etc).