1. Introduction

While it's true that we are now in the so-called Information Age, it's good to keep in mind that there are still many industries in which computers do not play a dominant role in day-to-day processes. Typically, these industries have a very long tradition, with processes and techniques that are passed on from masters to apprentices, generation after generation. However, as a reality check, unless it's a completely niche industry, chances are that companies who do not embrace new technologies will find themselves with a huge competitive disadvantage against competitors who have done so. We believe that by implementing a new order processing system for XYZ Printing Co. we can help propel the company into the 21st century without interfering with company values and other long standing traditions of the trade; but rather enhancing their ability to grow and adapt into this new way of doing business on a global scale.

1.1 Purpose of the system

To facilitate and automate production for XYZ printing Co. To provide a uniform interface for customer order submission, employee workflow, and management activities - this will enhance productivity and efficiency.

1.2 Scope of the system

The system will consist of a web based front-end for customers place orders and track progress; and a back end to allow managers and workers to receive, organize and schedule customer orders for production. Billing will not be within the initial scope of the system, however, the system shall be easily extensible to support future credit / debit features.

1.3 Objectives and success criteria of the project

To centralize and streamline order entry and processing. Success Scenario: Order entry is done solely by customers, without need to call the print company. No orders are lost due to human error during processing. Production times should be reduced from 3 to 1 business days.

1.4 Definitions, acronyms, and abbreviations

PWAS: Printshop Workflow Automation System, the name of the proposed system.

1.5 References

This system does not reference any other system.

1.6 Overview

This is a custom-built system, specifically to meet XYZ Printing Company's needs. It will be customized to help the company manage incoming orders and enhance production, thus saving the cost of hiring additional personnel. In the next sections, the challenges faced by the company will be explained in more detail, along with the proposed solutions.

2. Current system

Orders are submitted by customers, using various methods. For example, one customer might submit an order by email, another may submit one in person, and a third may choose to use the postal service. Employees receive individual orders and create corresponding job tickets based on the customer's requirements. New orders are printed and placed in a folder, where plant managers will physically sort and aggregate them, according to size, quantities and stocks to be used. These aggregations are then run in gang-style print runs, where many orders are processed simultaneously.

3. Proposed system

3.1 Overview

Printshop Workflow Automation System ("PWAS") is a web-based order taking and tracking portal. It will allow customers to place and track orders online, providing them updates at each production milestone. The system will allow employees to organize customer orders into print runs and track their completion status. System administrators will be able to manage existing user accounts or add new ones. User-access control will be provided to differentiate views of the system between customers, employees, and administrators.

3.2 Functional requirements

The proposed system shall provide the following major characteristics:

- The system shall allow customers to place and track orders, utilizing a payment method of their choice.
- The system shall allow customers to view their order history and account information.
- The system shall allow employees to organize, track and complete customer orders.
- The system shall allow administrators to manage user accounts, customer orders and printing properties.
- The system shall have user access control for security and access differentiation.

3.3 Nonfunctional requirements

3.3.1 Usability

The user interface should be understandable to non-technical customers, allowing them to submit, view, and edit orders. The logo should not have any religious, political, racist, sexual, or discriminatory connotations. Fonts should be clear and easy to read. Color scheme should be light background with dark foreground, to maximize contrast. There will be various help options for customers that explain the order submission and tracking processes.

3.3.2 Reliability

The system should be highly available, with 99% up time. Maintenance should not be required more than once a month.

3.3.3 Performance

The system will respond within thirty seconds for any user action, including work-order submission, order tracking, and any other user interaction with the system. The system should be available during business hours 99% of the time, with downtime allowed as specified by Section 3.3.2.

3.3.4 Supportability

The system will not interfere with previously created orders or with the history of previous transactions. The existing process for ordering will be supported by the system via a customer service employee, who will act as a proxy for offline customers. System maintenance should handle all updates required to fix defects, or handle change requests. The system will be available only in English. The system is web-based, so it is compatible with any operating system that can run a supported web-browser and connect to the Internet.

3.3.5 Implementation

The system will be web-based. It will support Internet Explorer 7+ and Firefox 3+. It should be implemented in a programming language that is cross-platform, so no porting will be required to change platforms.

3.3.6 Interface

The system shall be extensible to interface with a credit card processing service in the future. This functionality is not within the current scope of the system, as defined in Section 1.2.

3.3.7 Packaging

Personalized installation/configuration will be offered by the software company. The product should be hosted internally by the print shop.

3.3.8 Legal

The system does not have any legal requirements. No government or security clearance is necessary. The system is not implemented to comply with any particular disability users may have. Reasonable measures will be taken to protect private customer information, such as order history or account information.

3.4 System models

3.4.1 Scenarios

Scenario Name	register
Participating actor instances	bob: Customer
Flow of events	1. Bob selects the "Register" function on the PWAS website. 2. PWAS responds by displaying a form containing all information needed to register a new user. 3. Bob enters his full name, username, password, email address, and home address, then submits the form. 4. PWAS responds by confirming Bob's choice, creating his account and emailing Bob with his account information.

Scenario Name	login
Participating actor instances	bob: Customer
Flow of events	1. Bob selects the "Login" function on the PWAS website. 2. PWAS responds by displaying a form with username and password fields. 3. Bob enters his username and password, but makes a mistake, then submits the form. 4. PWAS responds by telling Bob that his username / password combination is invalid. 5. Bob corrects his mistake and submits the form. 6. PWAS responds by granting Bob access to the system.

Scenario Name	logout
Participating actor instances	bob: Customer
Flow of events	1. Bob selects the "Logout" function on the PWAS website. 2. PWAS responds by logging Bob out of the system.

Scenario Name	editProfile
Participating actor instances	bob: Customer
Flow of events	1. Bob selects the "Edit Profile" function on the PWAS website. 2. PWAS responds by displaying a form containing all information already stored for Bob. 3. Bob decides to change his address and enters a new address into the form. 4. Bob then saves the profile changes.

Scenario Name	customerOrdering
Participating actor instances	alice: Customer
Flow of events	1. Alice logs into the system and selects the "Create Order" function on the PWAS website. 2. PWAS responds by displaying a form containing all the specifications of an order. 3. Alice fills out the form with all relevant details. PWAS responds by confirming Alice's choices and asking her whether she wants to pay the order now, or save it to pay later. 4. Alice chooses to pay the order later, so her order is saved and she is redirected into the Payment function of PWAS. 5. Later, Alice logs in and selects to submit and pay for her saved order, but makes a mistake when filling out her billing information. 6. PWAS responds by notifying Alice that her billing information is invalid, and asking her to check for errors. 7. Alice corrects the mistake and resubmits the order form. 8. PWAS responds by confirming her order, submitting her order, and processing the payment. 9. Later, Alice logs in and selects the "Order Tracking" function of PWAS. 10. PWAS responds by displaying a list of all her orders, including previously saved orders and submitted orders. 11. Alice selects her most recent order. 12. PWAS responds by displaying all relevant details of her order.

Scenario Name	customerService
Participating actor instances	bob:CustomerService kimi:OfflineCustomer
Flow of events	1. Bob receives a phone call from Kimi who wishes to place an order. 2. Bob logs into the system and selects the "Create Order" function on the PWAS website. 3. PWAS responds by displaying a form containing all relevant specifications of an order, to be filled out by Bob. 4. Bob fills out the form, according to Kimi's requirements, by selecting a typical business flyer, the type of paper, and color specifications. 5. After creating the order, PWAS asks Bob whether he wants to pay the order now, or save it to pay later. 6. Kimi would like to pay later, so Bob chooses to save the order without submitting it to be processed. 7. PWAS responds by confirming that the order is saved. 8. Kimi asks Bob for the status of a previous order. 9. Bob selects the "Order Info" function on the PWAS website. 10. PWAS responds with a list of all Kimi's orders. 11. Bob selects Kimi's previous order. 12. PWAS responds by displaying all relevant details of the past order.

Scenario Name	userAdministration
Participating actor instances	john:Administrator
Flow of events	1. John logs into the system and selects the "View User Account" function on the PWAS website. 2. PWAS responds by displaying a list of all users registered with the system. 3. John chooses a particular user account. 4. PWAS responds by displaying a detailed, read-only, summary of that account. 5. John reviews the information, then selects the "Edit User Account" function on the PWAS website. 6. PWAS responds by making the user account editable. John changes the user's address and saves his changes. 7. PWAS responds by confirming John request and updating the user's record. 8. John then selects a former employee and selects the "Delete Account" function on the PWAS website. 9. PWAS responds by confirming John request and deleting the user's record.

Scenario Name	orderAdministration
Participating actor instances	john:Administrator
Flow of events	1. John logs into the system and selects the "View Orders" function on the PWAS website. 2. PWAS responds by displaying a list of all orders contained within the system. 3. John selects a particular customer order. 4. PWAS responds by displaying a detailed, read-only, summary of that order. 5. John reviews the information, then selects the "Edit Order Status" function on the PWAS website. 6. PWAS responds by making the order editable. 7. John changes the order's status and saves his changes. 8. PWAS responds by confirming John's request and updating the order's record. 9. John then selects a canceled order and selects the "Delete Order" function on the PWAS website. 10. PWAS responds by confirming John's request and deleting the canceled order.

Scenario Name	preprinting
Participating actor instances	alice, bob:Employee
Flow of events	1. Alice logs into the system and selects the "View Work Pool" function on the PWAS website. 2. PWAS responds by displaying a view of all customer orders that have yet to be sorted into a print run, along with relevant details. 3. Alice selects an order to examine in further detail. PWAS responds by displaying a detailed view of that specific order. 4. Alice returns to the list of all customer orders yet to be sorted into a print run. 5. PWAS responds by displaying a view of all customer orders that have yet to be sorted into a print run. 6. Alice selects the "Create Print Run" function of PWAS. 7. PWAS responds by creating an empty print run and notifying Alice. 8. Alice selects the "Edit Run" function of PWAS, with the new print run selected. 9. PWAS responds by showing Alice a form with options to add / remove orders to the print run. 10. Alice adds five orders to the print run. 11. PWAS responds by updating the status of the print run and notifying Alice. 12. Alice selects the "Submit Run To Printing" function of PWAS. 13. PWAS confirms Alice's choice, then finalizes the changes to the new print run, updates its status, and notifies the proper employees that a new job is ready for printing.

Scenario Name	printing
Participating actor instances	bob: Customer
Flow of events	1. Bob logs into the system and selects the "Printing" function on the PWAS website. 2. PWAS responds by displaying a list of orders that are ready for printing. 3. Bob selects an order to be printed, according to the queue, prints the order, and submits the information to PWAS. 4. PWAS responds by confirming Bob's choice and updates the print run's status to "Printed".

Scenario Name	finishing
Participating actor instances	bob: Customer
Flow of events	1. Bob logs into the system and selects the "Finishing" function on the PWAS website. 2. PWAS responds by displaying a list of orders that are ready for finishing. 3. Bob selects an order to be finished, according to the queue, finishes the order, and submits the information to PWAS. 4. PWAS responds by confirming Bob's choice and updates the print run's status to "Finished".

Scenario Name	shipping
Participating actor instances	bob: Customer
Flow of events	1. Bob logs into the system and selects the "Shipping" function on the PWAS website. 2. PWAS responds by displaying a list of orders that are ready for shipping. 3. Bob selects an order to be shipped, according to the queue, ships the order, and submits the information to PWAS. 4. PWAS responds by confirming Bob's choice and updates the print run's status to "Shipped".

3.4.2 Use Case Model

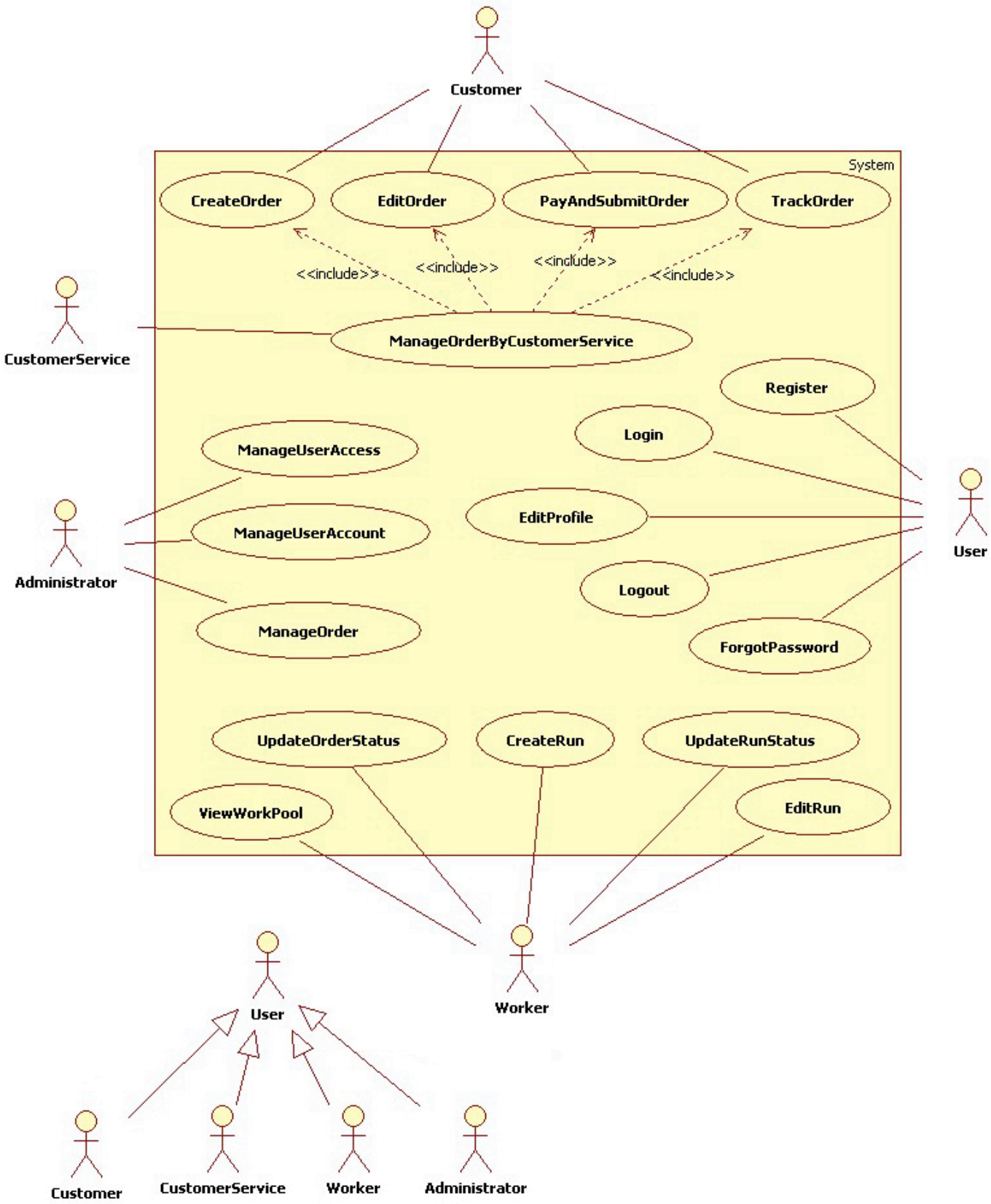

Figure 1 – Use Case Diagram for PWAS

Use case Name	Register
Participating Actors	User
Flow of Events	1. User activates the "Register" function of the system (i.e. clicks the “Register” link) [*1]. 2. The System responds with a form for the User to fill out 3. User provides required information (Full Name, Username, Password, Email address, home address, etc…) 4. User submits form 5. User receives a confirmation message that his account is created
Entry Conditions	• User is provided with a “Register” link [*2].
Exit Conditions	• User has a working username / password combination to login to the system
Exception	• Email Password is not valid • Password does not meet security requirements • Required information is missing from the form

*1 - Anytime we say: “activates the X function”, we mean “the user clicks on the X link to activate the functionality”.

*2 – It is assumed that the user is always provided the link he is supposed to click on.

Use case Name	Login
Participating Actors	User
Flow of Events	1. User activates the "Login" function of the system. 2. The System responds with a form for the User to fill out 3. User provides correct username and password 4. User submits login form
Entry Conditions	• User is registered in the System
Exit Conditions	• User is authenticated in the System • User is redirected to the home page
Exception	• Username or password field is left empty. • Incorrect username or password is entered.

Use case Name	Logout
Participating Actors	User
Flow of Events	1. User activates the "Logout" function of the system. 2. The system logs the user out.
Entry Conditions	• User is logged into the System[*3]
Exit Conditions	• User is logged out of the System • User is redirected to the home page
Exception	• No Exceptions

*3 – We assume that all Use Cases (except Register, Login, and Forgot Password) have this requirement.

Use case Name	ForgotPassword
Participating Actors	User
Flow of Events	1. User activates the "Forgot Password" function of the system. 2. The Systems prompts the User for a username 3. User enters valid username 4. User submits form
Entry Conditions	• User has a valid login
Exit Conditions	• System sends the password to the email address stored for the username entered • System displays status message
Exception	• The username entered is not found in the System

Use case Name	EditProfile
Participating Actors	User
Flow of Events	1. User activates the "Agent Profile" function of the system. 2. The System responds with a form containing the existing User Information 3. User makes changes to any of the information fields 4. User submits the updated information to the System.
Entry Conditions	• User is logged in.
Exit Conditions	• Information is updated into the User's profile.
Exception	• The new information entered does not pass validation

Use case Name	CreateOrder
Participating Actors	Customer, CustomerService
Flow of Events	1. Customer activates the "Create Order" function of the system. 2. The System responds by showing the "Create Order Form" 3. Customer fills out the order form by selecting all the specs of the specific item he wants to order, and uploads the file that will be printed. 4. After filling out the form the Customer saves the Order into the System.
Entry Conditions	• Customer is logged into the System
Exit Conditions	• The Customer has created an order into the System.
Exception	• No Exceptions

Use case Name	EditOrder
Participating Actors	Customer, CustomerService
Flow of Events	1. Customer activates the "Edit Order" function on the System 2. The System responds by showing the Customer the list of Orders that this Customer has created. 3. Customer changes details about the Order, such as the quantity to print, or the paper type. 4. After filling out the changes the Customer saves them into the System.
Entry Conditions	• Customer is logged into the System • Order has been created, saved, and has not been paid for.
Exit Conditions	• The Customer has edited some information about the Order.
Exception	• No Exceptions

Use case Name	PayAndSubmitOrder
Participating Actors	Customer, CustomerService
Flow of Events	1. The Customer goes to the Manage Orders page of the system. 2. System shows the Customer a list of Orders, and each Order has an option to pay for it (if it hasn't been paid for). 3. The Customer clicks the Pay Now button, pays for the order, and submits the payment. 4. The System validates the Payment information to make sure it's correct. 5. After the payment is accepted, the Order is submitted into the System as Paid and is now available to be processed.
Entry Conditions	• An Order has been selected for payment
Exit Conditions	• The order has been paid and submitted for processing.
Exception	• If the payment information that the Customer entered is not correct, the System will let Customer know, and allow Customer to re-enter this information

Use case Name	TrackOrder
Participating Actors	Customer, Customer Service
Flow of Events	1. Customer activates the "Order Tracking" function of the System. 2. The System shows the Customer a list of orders belonging to this Customer. 3. The Customer selects an Order. 4. The System returns to the Customer the details about the Order.
Entry Conditions	• The Customer initiating function already has orders in the System
Exit Conditions	• The Customer sees the Order Tracking information
Exception	• The Customer doesn't have any order on the System.

Use case Name	ManageUserAccount (View)
Participating Actors	Administrator
Flow of Events	1. Administrator activates "Manage User Account" function 2. The System responds by displaying a list of System users 3. Administrator selects a System user to see their information 4. The System displays the User's information 5. Administrator checks the information
Entry Conditions	• Administrator is logged into the System
Exit Conditions	• Administrator has viewed the information
Exception	• No Exceptions

Use case Name	ManageUserAccount (Edit)
Participating Actors	Administrator
Flow of Events	1. The Administrator activates " Manage User Account" function 2. The System responds by displaying a list of System users 3. The Administrator selects an entry from the list, edits the information and submits it. 4. The System responds by confirming the modifications 5. The Administrator confirms the information 6. The System updates the System user's information
Entry Conditions	• Administrator is logged into the System
Exit Conditions	• The information is successfully updated
Exception	• Administrator cancels the edit process

Use case Name	ManageUserAccess
Participating Actors	Administrator
Flow of Events	7. The Administrator activates " Manage User Access" function 8. The System responds by displaying a list of System users 9. The Administrator edits the user's role and submits the form. 10. The System responds by confirming the modifications 11. The Administrator confirms the information 12. The System updates the System user's information
Entry Conditions	• Administrator is logged into the System
Exit Conditions	• The information is successfully updated
Exception	• Administrator cancels the edit process

Use case Name	ManageOrder
Participating Actors	Administrator
Flow of Events	1. The Administrator activates “Manage Order” function 2. The System responds by displaying a list of current orders that satisfy the Entry Condition. 3. The Administrator selects an order to edit details for. 4. The System responds by displaying the order's information 5. The Administrator changes some Order details, like quantity, or paper type. 6. The Administrator then saves the changes into the System.
Entry Conditions	• Administrator is logged into the System • An Order has been paid for by a Customer, but has not been added to a Print Run by a Worker.
Exit Conditions	• Administrator has changed Order details.
Exception	• No Exceptions

Use case Name	ViewWorkPool
Participating Actors	Worker
Flow of Events	1. Worker activates the “View Orders” function of the System 2. The System responds by presenting a view of all available customer orders to fill. 3. The Worker may select an order to see details of a specific order. 4. The System responds by showing all details of a specific order
Entry Conditions	• Worker is logged into the System
Exit Conditions	• The Worker has completed viewing the available customer orders
Exception	• No exceptions

Use case Name	CreateRun
Participating Actors	Worker
Flow of Events	1. Worker activates the "Create Print Run" function of the System 2. The System responds by showing the Worker a form to be filled out. 3. The worker fills out the form with the Print Run details and clicks Submit. 4. The system responds by creating a Print Run in the Created state, and notifying the Worker.
Entry Conditions	• Worker is logged into the System
Exit Conditions	• The Worker has completed creating a print run.
Exception	• No exceptions

Use case Name	EditRun
Participating Actors	Worker
Flow of Events	1. The Worker initiates the "Edit Run" function 2. The System responds by showing the Worker a form with options to add orders to the print run. 3. The Worker adds orders to the print run and clicks Submit. 4. The System updates the status of the print run from Created to PrePrinting.
Entry Conditions	• Worker is logged into the System • Worker has selected an existing print run to edit, which is currently in the “Created” state.
Exit Conditions	• A print run has been edited and saved.
Exception	• No exceptions

Use case Name	UpdateRunStatus (Printing)
Participating Actors	Worker
Flow of Events	1. The Worker activates the "Update Run Status" function 2. The System shows the Worker a list of Runs 3. The Worker selects a Run that’s currently in the “PrePrinting” state. 4. The Worker then changes the Run status from “PrePrinting” -> “Printing” 5. The new Run status is saved into the System.
Entry Conditions	• Worker is logged into the System • The System contains Runs in the “PrePrinting” state.
Exit Conditions	• The print run's status has been updated to reflect that it's now in the Printing phase
Exception	• No exceptions

Use case Name	UpdateRunStatus (Finishing)
Participating Actors	Worker
Flow of Events	1. The Worker activates the "Update Run Status" function 2. The System shows the Worker a list of Runs 3. The Worker selects a Run that’s currently in the “Printing” state. 4. The Worker then changes the Run status from “Printing” -> “Finishing” 5. The new Run status is saved into the System.
Entry Conditions	• Worker is logged into the System • The System contains Runs in the “Printing” state.
Exit Conditions	• The print run's status has been updated to reflect that it's now in the Finishing phase
Exception	• No exceptions

Use case Name	UpdateRunStatus (Shipping)
Participating Actors	Worker
Flow of Events	1. The Worker activates the "Update Run Status" function 2. The System shows the Worker a list of Runs 3. The Worker selects a Run that's currently in the "Finishing" state. 4. The Worker then changes the Run status from "Finishing" -> "Shipping" 5. The new Run status is saved into the System.
Entry Conditions	• Worker is logged into the System • The System contains Runs in the "Finishing" state.
Exit Conditions	• The print run's status has been updated to reflect that it's now in the Printing' phase
Exception	• No exceptions

Use case Name	UpdateRunStatus (Closed)
Participating Actors	Worker
Flow of Events	6. The Worker activates the "Update Run Status" function 7. The System shows the Worker a list of Runs 8. The Worker selects a Run that's currently in the "Shipping" state. 9. The Worker then changes the Run status from "Shipping" -> "Closed" 10. The new Run status is saved into the System.
Entry Conditions	• Worker is logged into the System • The System contains Runs in the "Shipping" state.
Exit Conditions	• The print run's status has been updated to reflect that it's now in the Closed phase
Exception	• No exceptions

Use case Name	UpdateOrderStatus [*4]
Participating Actors	Worker
Flow of Events	1. The Worker activates the "Update Order Status" function 2. The System shows the Worker a list of Runs 3. The Worker selects an Order and updates its status. 4. The new Order status is saved into the System.
Entry Conditions	• Worker is logged into the System • The System contains Orders.
Exit Conditions	• None
Exception	• No exceptions

*4 – This User Case (UpdateOrderStatus) with the Actor "Worker" is only present in this system for internal purposes, in case the Print Shop has to manually change the status of an order manually for whatever reason. In normal cases, the system will update the Order status automatically as the Print Run changes status.

3.4.3 Object Model

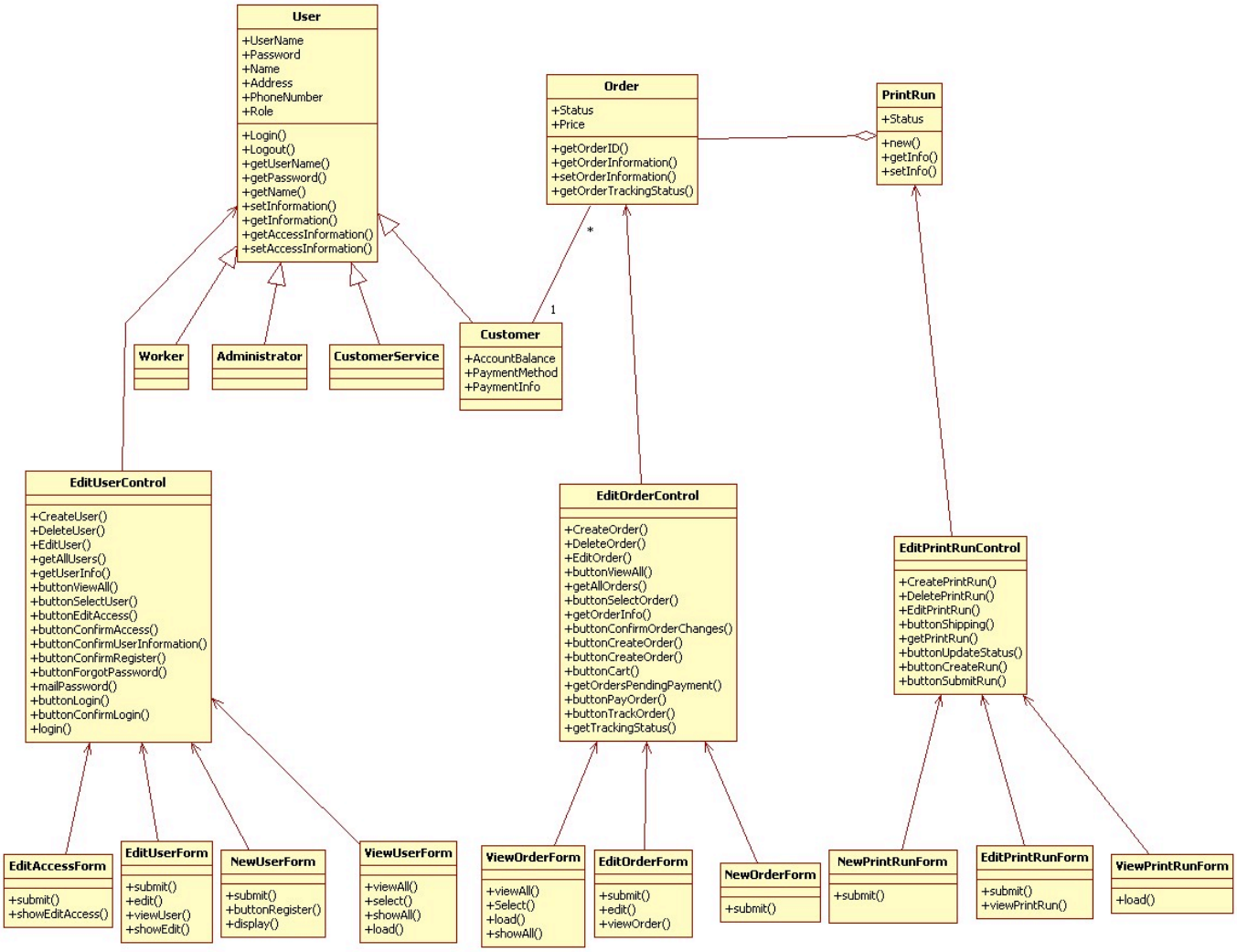

Figure 2 – Object Diagram for PWAS

3.4.4 Dynamic Model

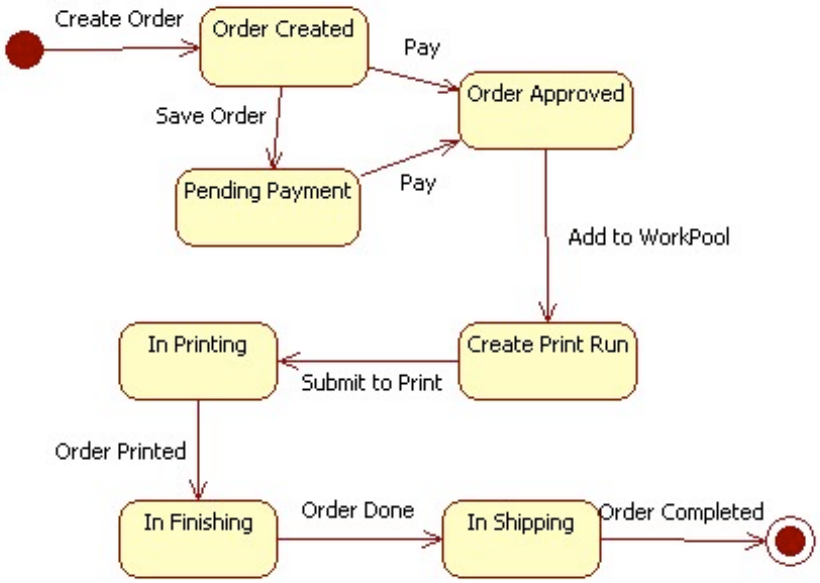

Figure 3 – Statechart Diagram for "Order"

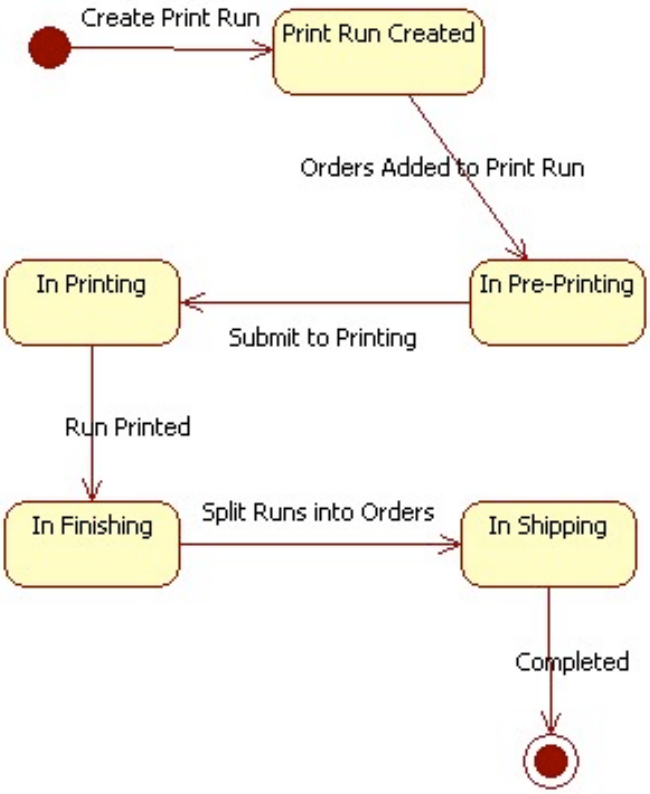

Figure 4 – Statechart Diagram for "Print Run"

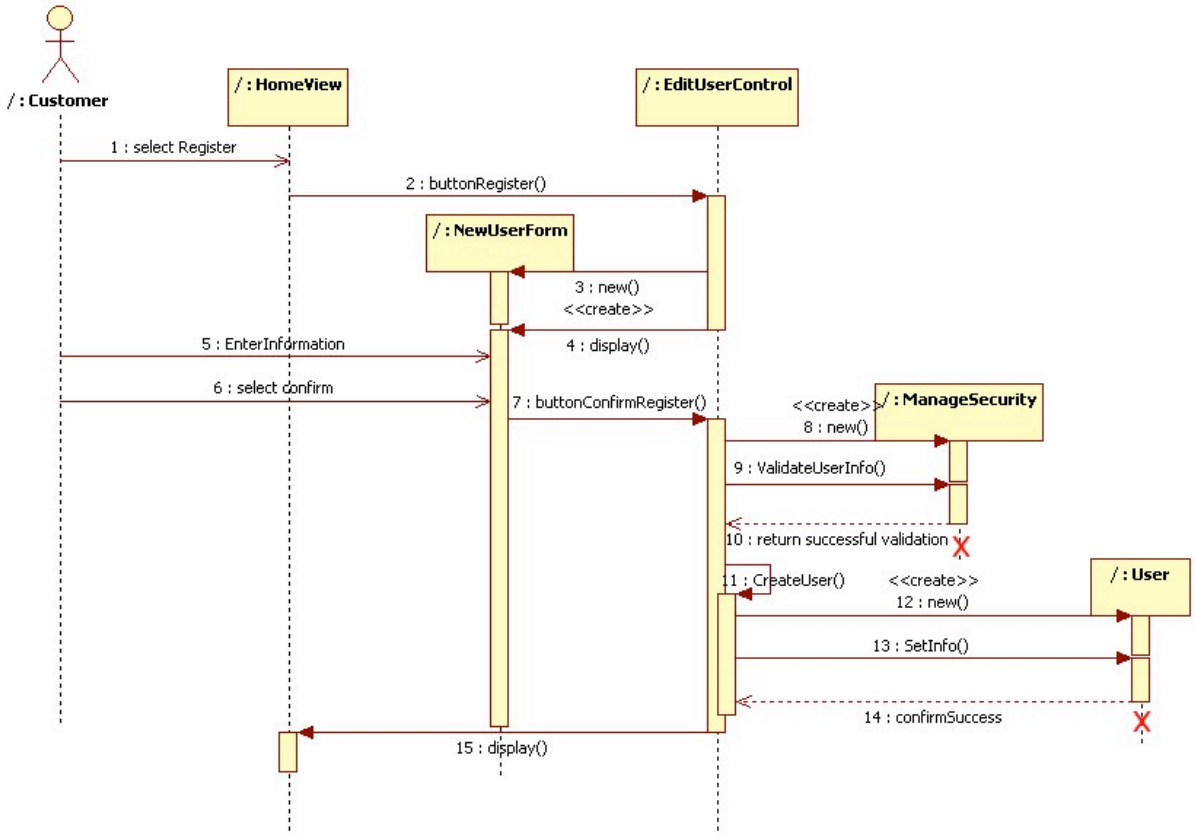

Figure 5 – Sequence Diagram for "Register"

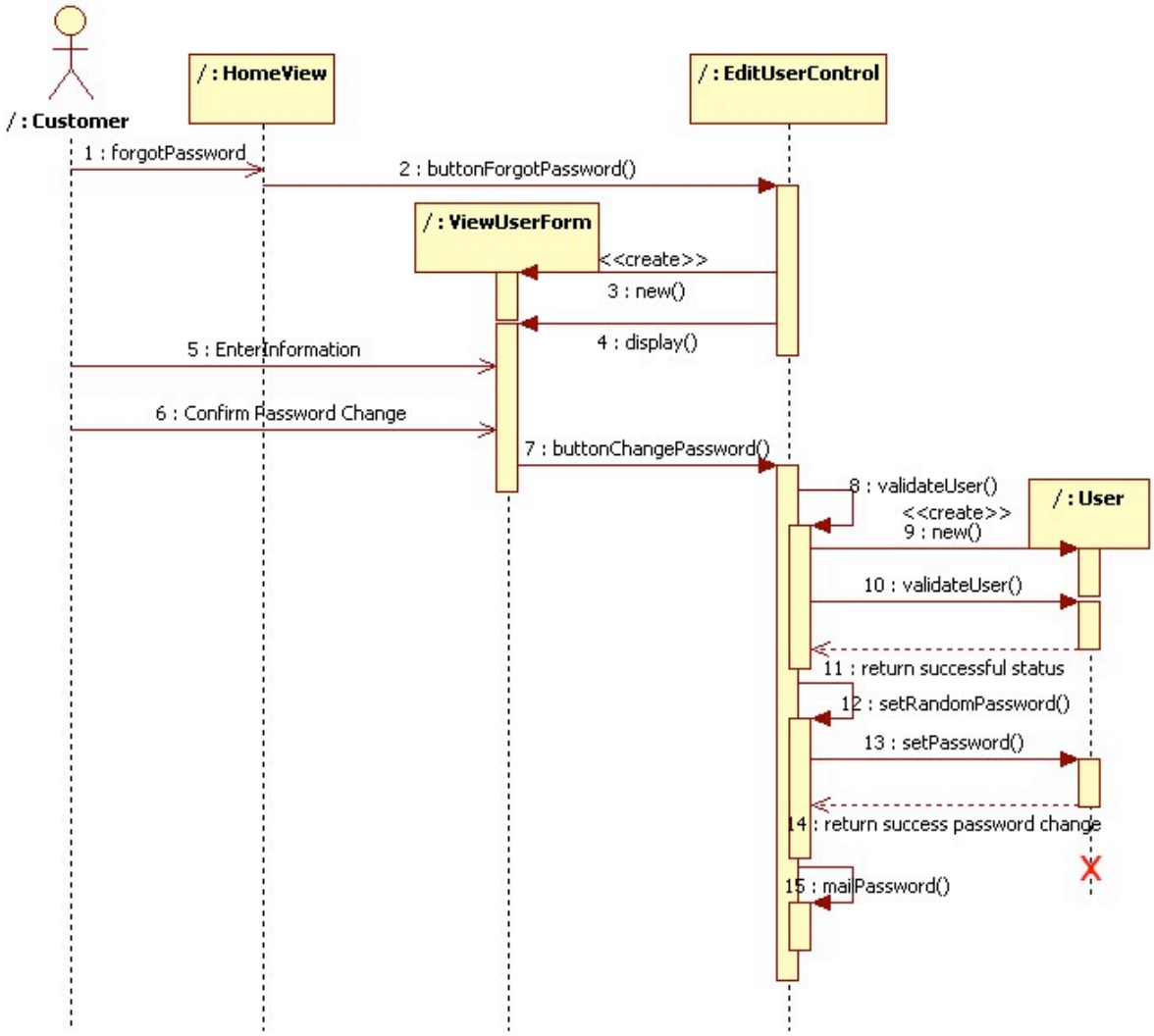

Figure 6 – Sequence Diagram for "Forgot Password"

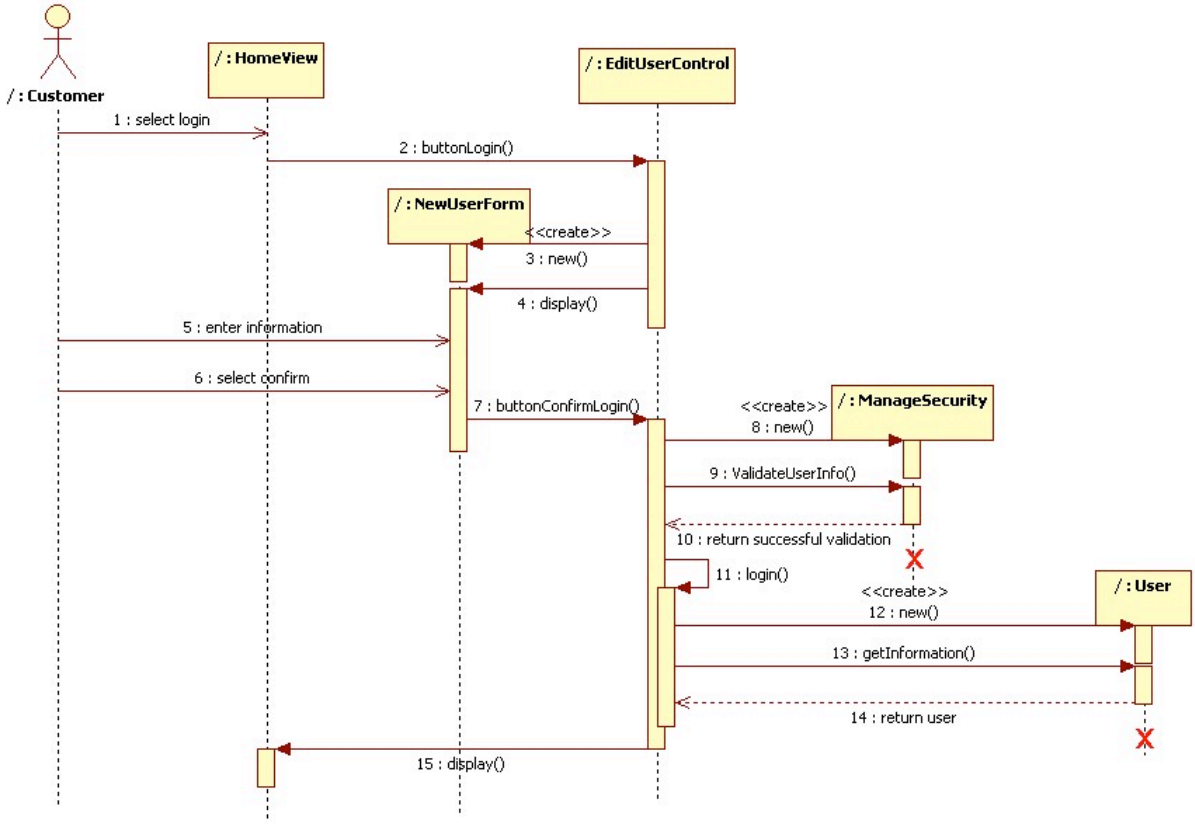

Figure 7 – Sequence Diagram for "Login"

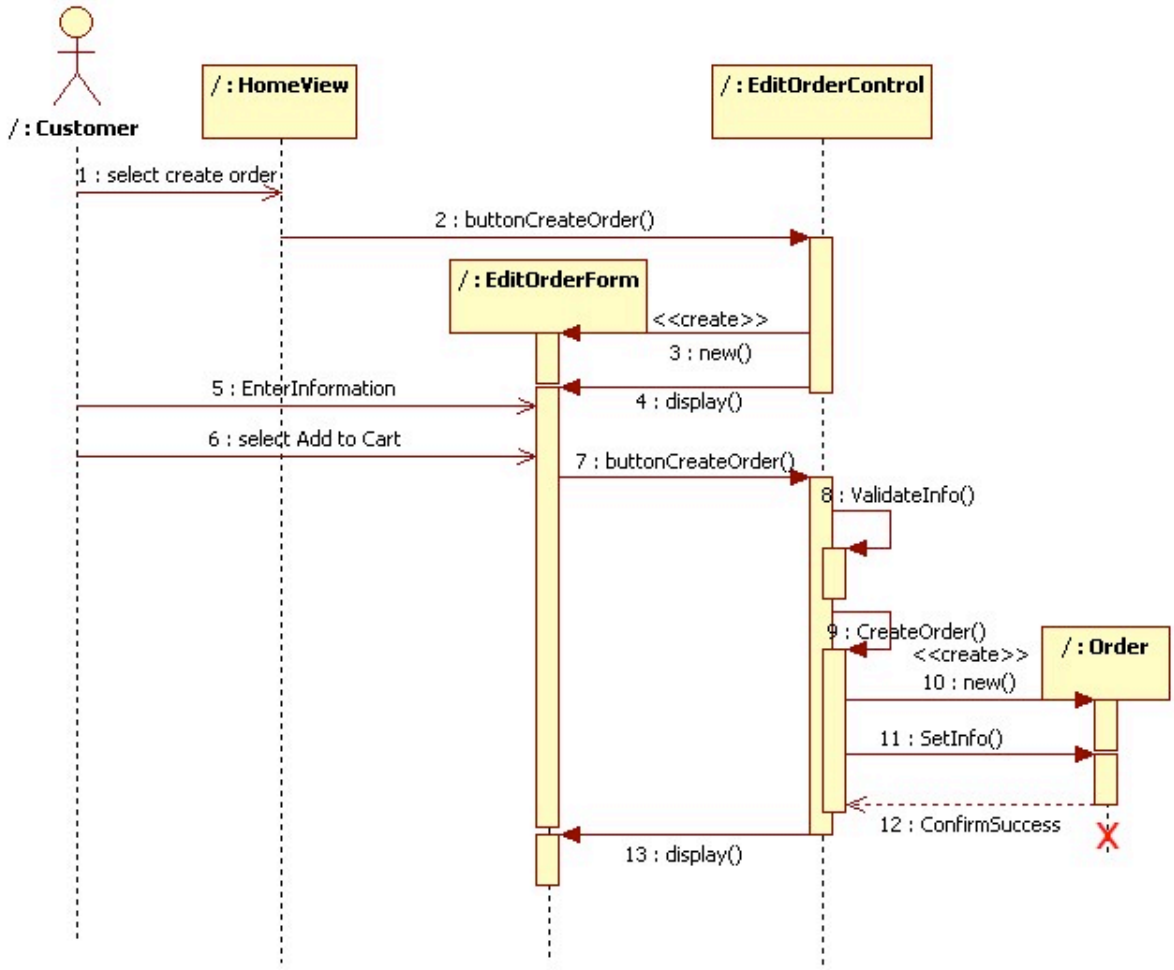

Figure 8 – Sequence Diagram for "Create Order"

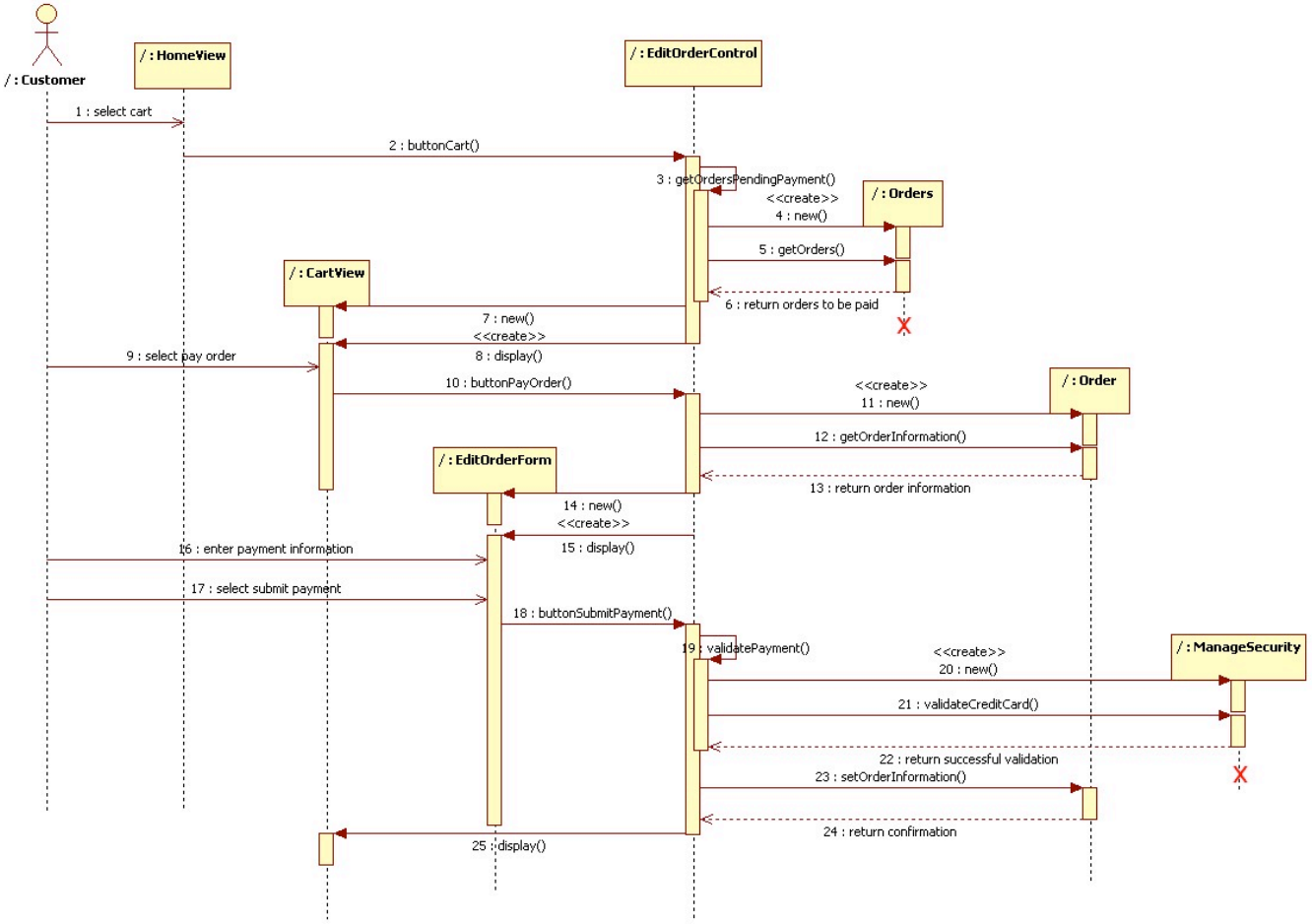

Figure 9 – Sequence Diagram for "Pay Order"

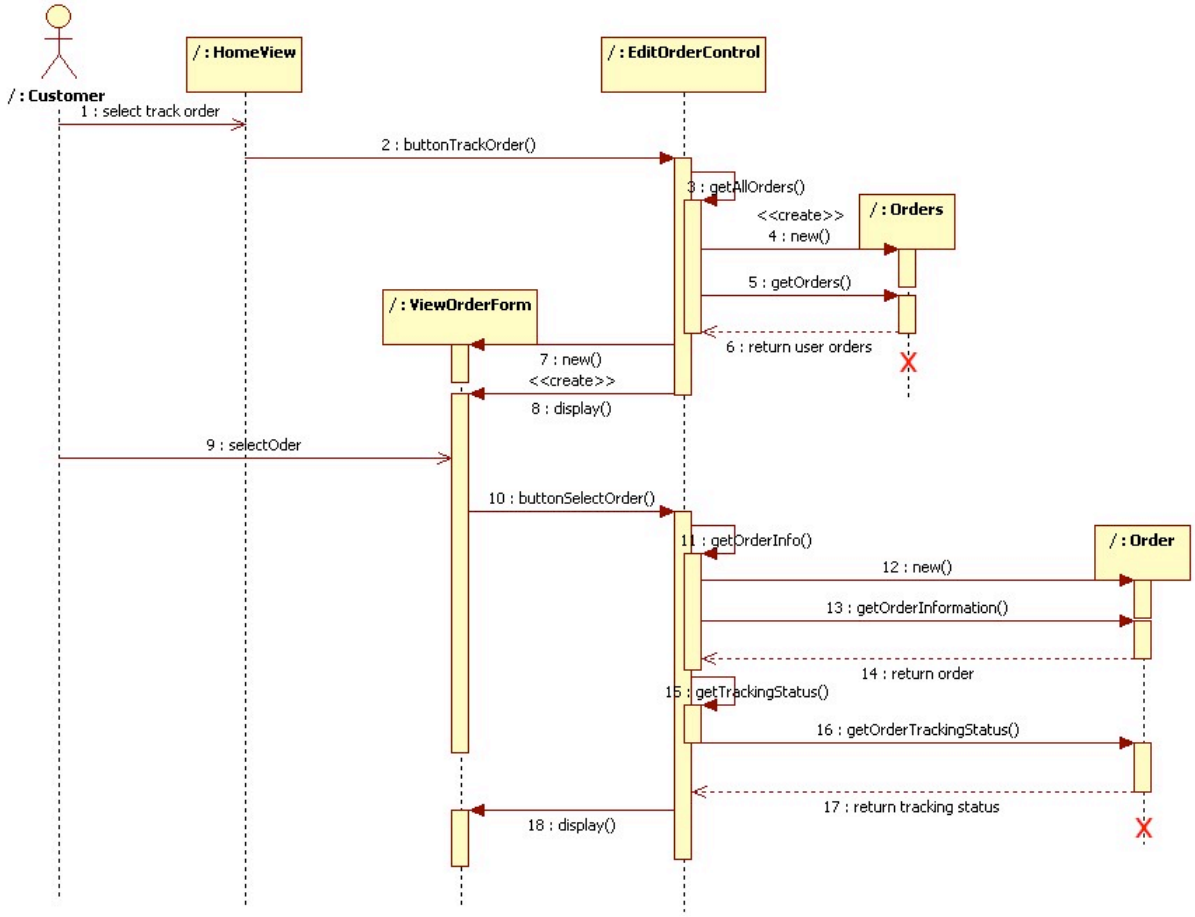

Figure 10 – Sequence Diagram for "Track Order"

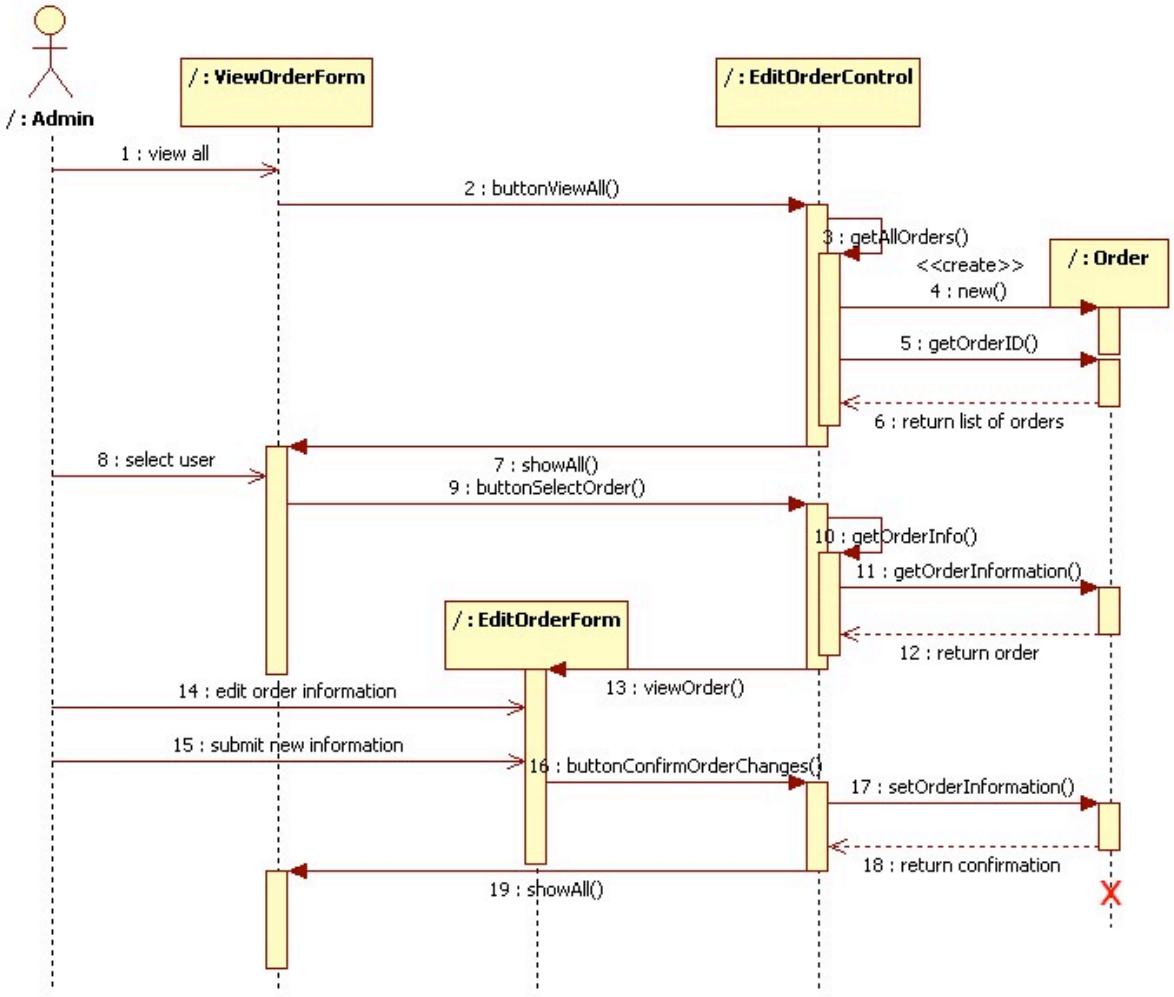

Figure 11 – Sequence Diagram for "Manage Order"

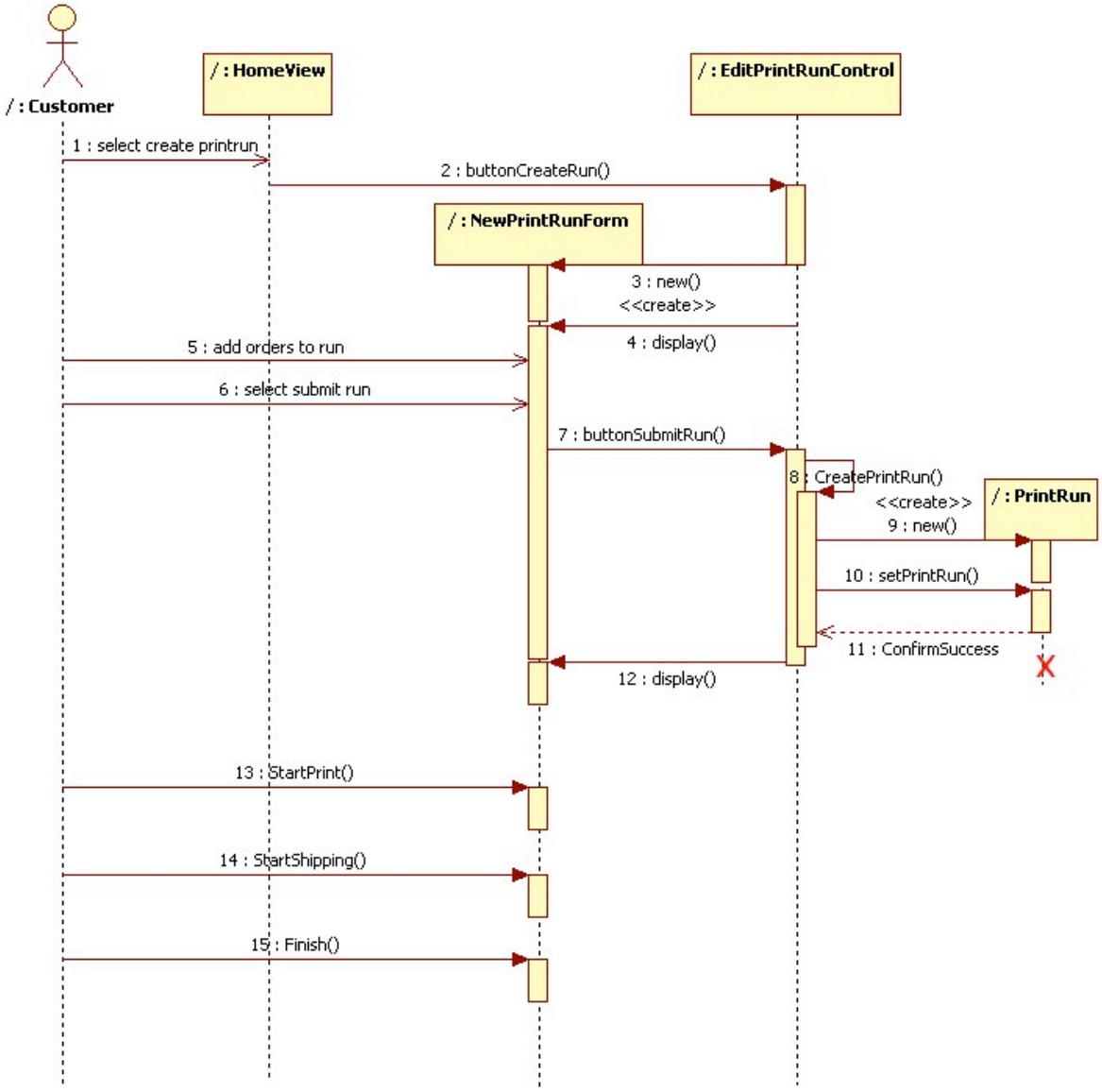

Figure 12 – Sequence Diagram for "Create Run"

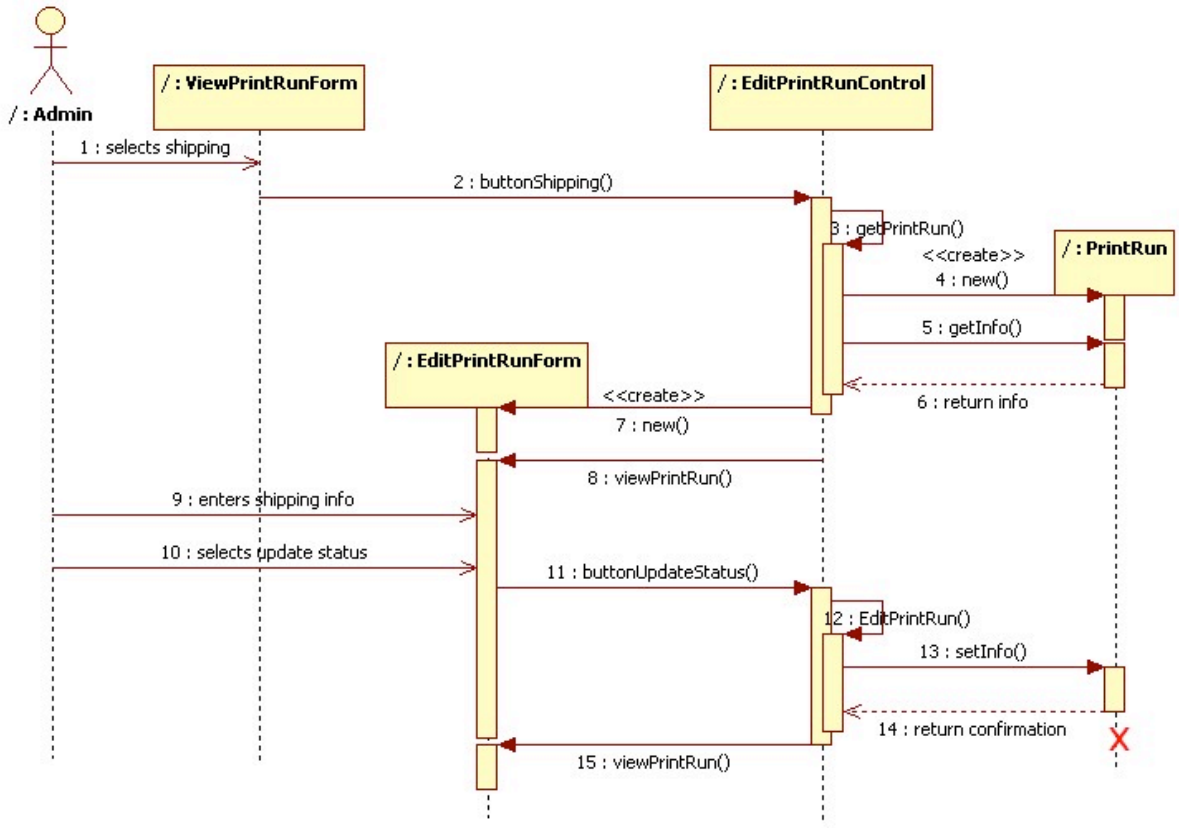

Figure 13 – Sequence Diagram for "Update Run Status (Shipping)"

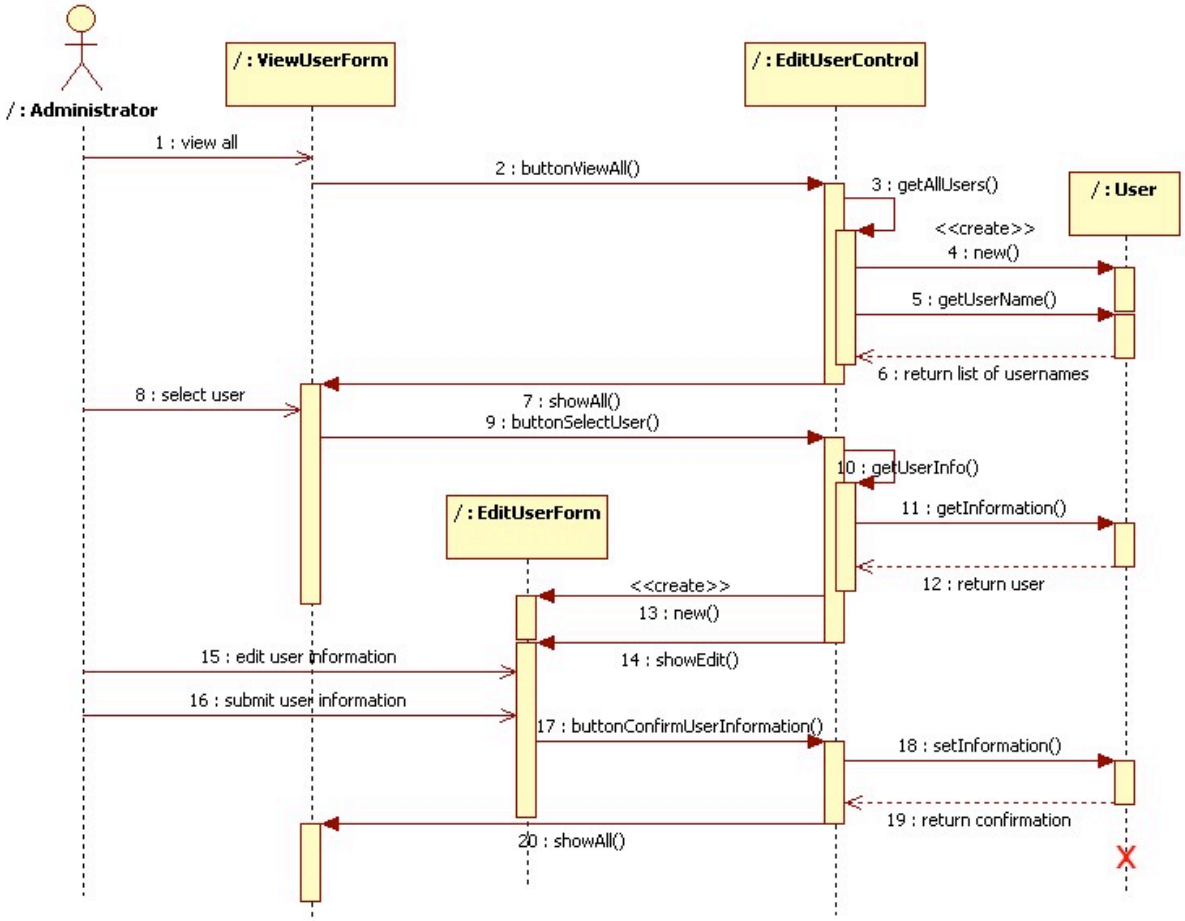

Figure 14 – Sequence Diagram for "Manage User Account (Edit)"

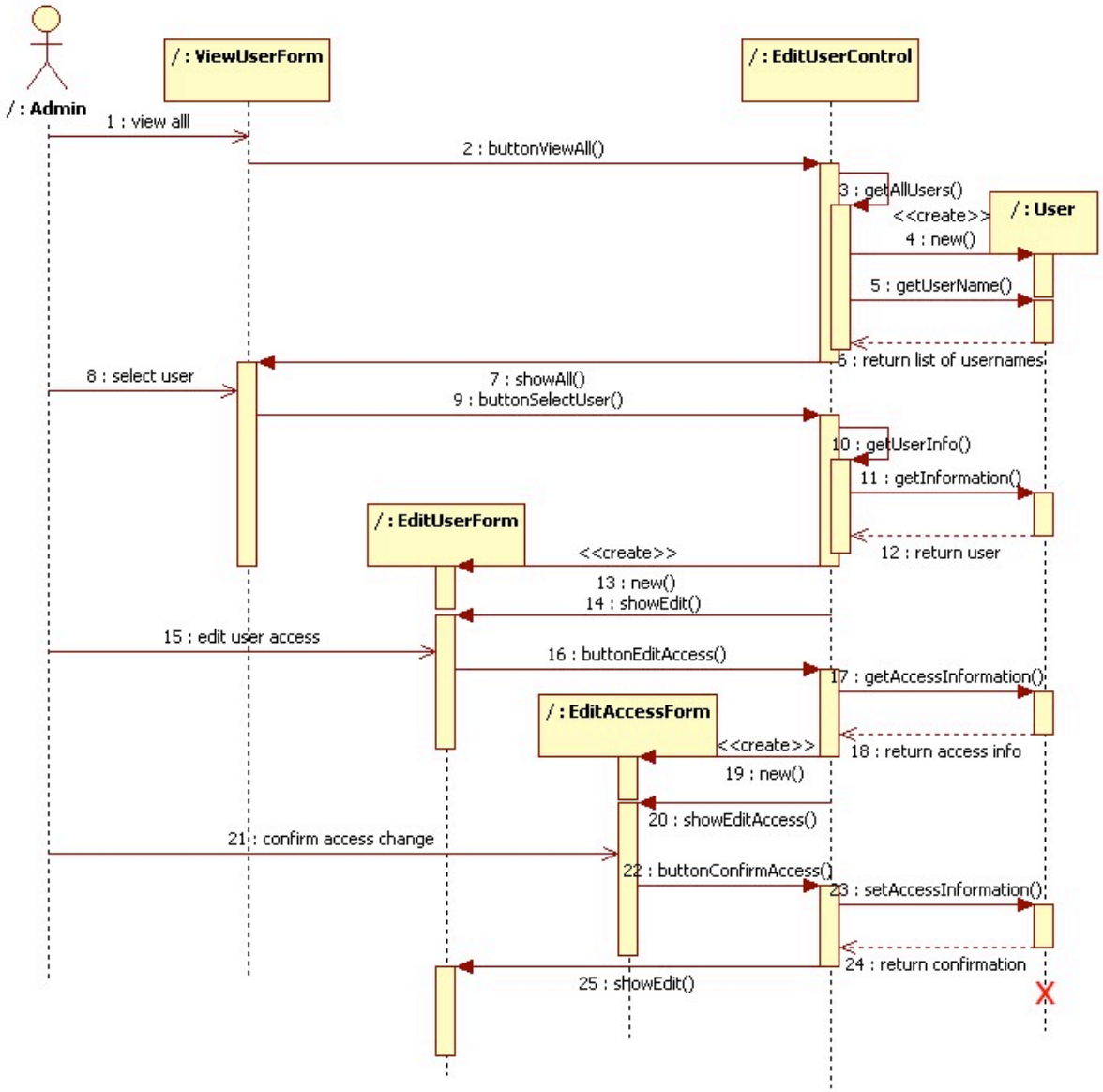

Figure 15 – Sequence Diagram for "Manage User Access"

3.4.5 User Interface

Menu

- Place an Order.
- View Order Status.
- Edit Profile.
- Search.

Company News!

Vestibulum sit amet nulla era

20th May 2009

Lorem ipsum dolor sit amet, consectetur.

20th May 2009

Lorem ipsum dolor sit amet.

Profile

Contact Information

First Name:	Masoud
Last Name:	Sadjadi
Company Name:	M.S.Soft Inc.
Email Address:	M.S@MSSoft.com
Phone #:	305-345-6321

Shipping Information

Address Line 1:	123 Main St.
Adress Line 2:	Suite 1023
City:	Miami Beach
State:	FL
Zip code:	33131

Billing Information

Credit Card #	3713 3452 2567 123
Card Type	Amex
Exp. Date	10/10
Security Code	5643
Name on Card	Masoud Sadjadi

Billing Address

Address Line 1:	123 Main St.
Adress Line 2:	Suite 1023
City:	Miami Beach
State:	FL
Zip code:	33131

Update Profile

Menu

- Place an Order.
- View Order Status.
- Edit Profile.
- Search.

Company

Order History

Job #	Job Name	Status	Date
34526	500 Business Cards	Received	10/05/2009
87623	1000 Flyers	Printing	10/04/2009
43353	5 Posters	Finishing	10/04/2009
12341	75 Invitations	Shipped	05/24/2009
45435	2000 Postcards	Shipped	01/04/2008

Menu

Search.

Company News!

Vestibulum sit amet nulla era

20th May 2009

Lorem ipsum dolor sit amet, consectetur.

Current Job

Components for run# 471

Job #	Job Name	Final size
34526	500 Business Cards	2 x 3.5
87623	1000 Flyers	8.5 x 11
43353	5 Posters	24 x 36
12341	75 Invitations	5 x 7
45435	2000 Postcards	4 x 6

Menu

Create a run.

Populate a run.

Search.

Company News!

Vestibulum sit amet nulla era

20th May 2009

Create Run

Run ID:

Run Size: 23 X 29

Run Qty: 500

Stock Finish: Glossy

Stock Weight Heavy

Create Run Clear

Menu

- Create a run.
- Populate a run.
- Search.

Company News!

Vestibulum sit amet nulla era

20th May 2009

Lorem ipsum dolor sit amet, consectetur.

20th May 2009

Lorem ipsum dolor sit amet, consectetur.

Incoming Orders

Move Selected Jobs to -> Run 471

Select	Job #	Job Name	Status	Date
☐	34526	500 Business Cards	Unassigned	10/05/2009
☐	87623	1000 Flyers	Unassigned	10/04/2009
☐	43353	5 Posters	Unassigned	10/04/2009
☐	12341	75 Invitations	Assigned (471)	05/24/2009
☐	45435	2000 Postcards	Assigned (471)	01/04/2008

Created Runs

Print selected

Select	Run ID	Run type	Status	Date
☐	471	500 23 x 29 Glossy Heavy	Open	10/05/2009
☐	470	1000 23 x 29 Glossy Light	Open	10/04/2009
☐	469	500 23 x 29 Matte Heavy	Open	10/04/2009
	468	200 23 x 29 Glossy Heavy	Printing	10/02/2009
	467	500 23 x 29 Glossy Heavy	Printing	10/01/2009

Menu

- Search.

Company News!

Vestibulum sit amet nulla era

20th May 2009

Current Job

Run ID	Run type	Status	Date
471	500 23 x 29 Glossy Heavy	Open	10/05/2009

4. Glossary

- **Administrator**: A member of the company, who has all the rights of a regular Employee plus other administrative rights such as deleting a user, editing a user's information, etc.
- **Customer**: A client of the company, who can submit orders for printing, pay those orders, and track the orders as well.
- **Company:** Specifically, XYZ Printing Co.
- **Customer Service:** A member of the company who can take an order on behalf of a customer – to act as a proxy for an offline customer.
- **Worker**: A member of the company, who has all the rights of any User plus other rights such as create print runs, add orders to print runs, etc.
- **Finishing:** The part of the company workflow where the cutting and resizing process is taking place.
- **Order**: A User can create an order and save it into the system, which contains specifications regarding printing details, a file to be printed, and payment information.
- **Portal**: Web-based interface presented to customer and employees.
- **Printing:** The part of the company workflow where the print-manufacturing process is taking place.
- **Print Run**: A single file created by an employee, which is sent to printing.
- **System**: PWAS is considered the system, and it entails all the software that takes care of the workflow management.