**State Model Process**

**A.**

**Development of possible Events**

1. Bet
2. Deal
3. Hit
4. Stay
5. Bust

**Determining States and Transitions for each Event**

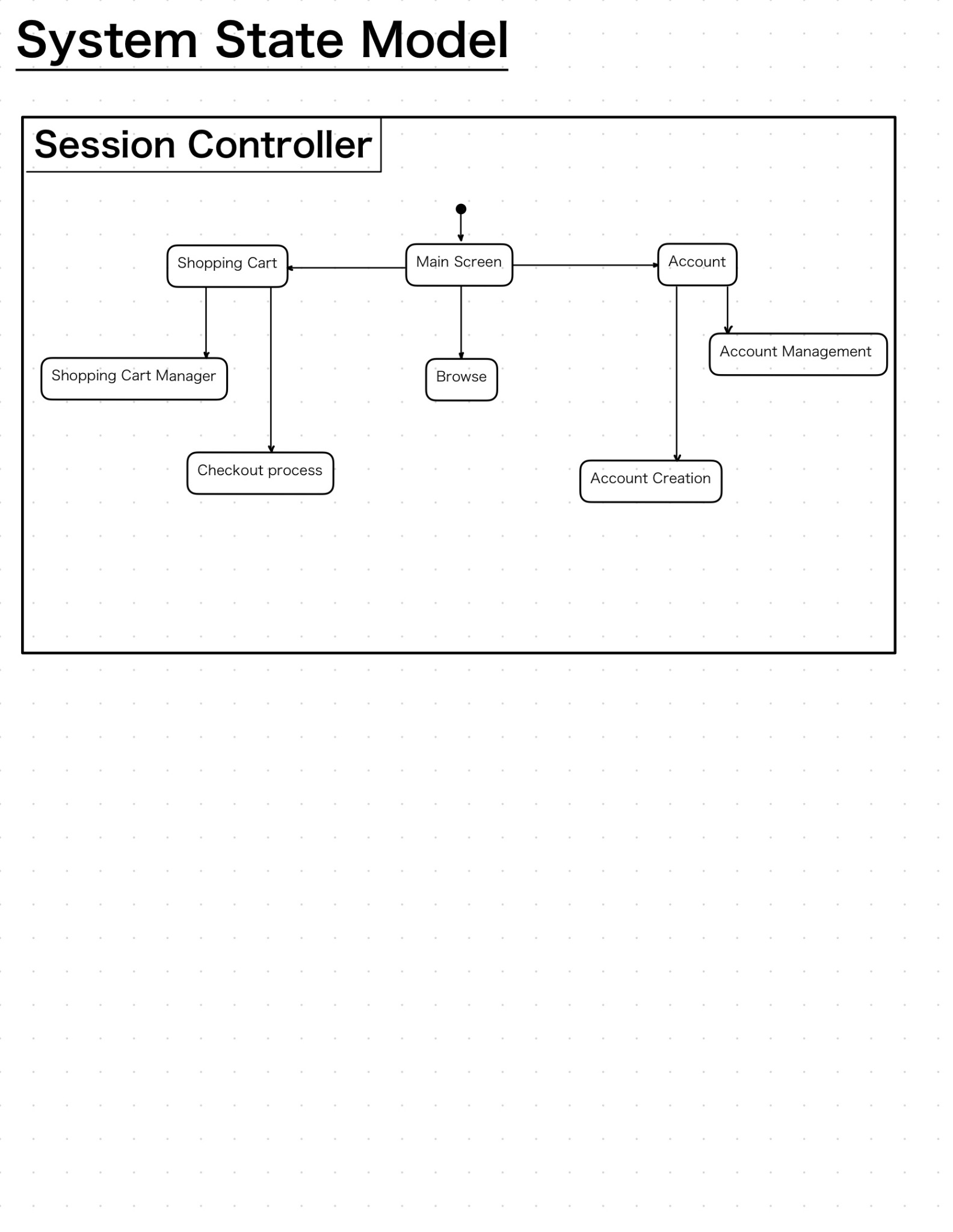
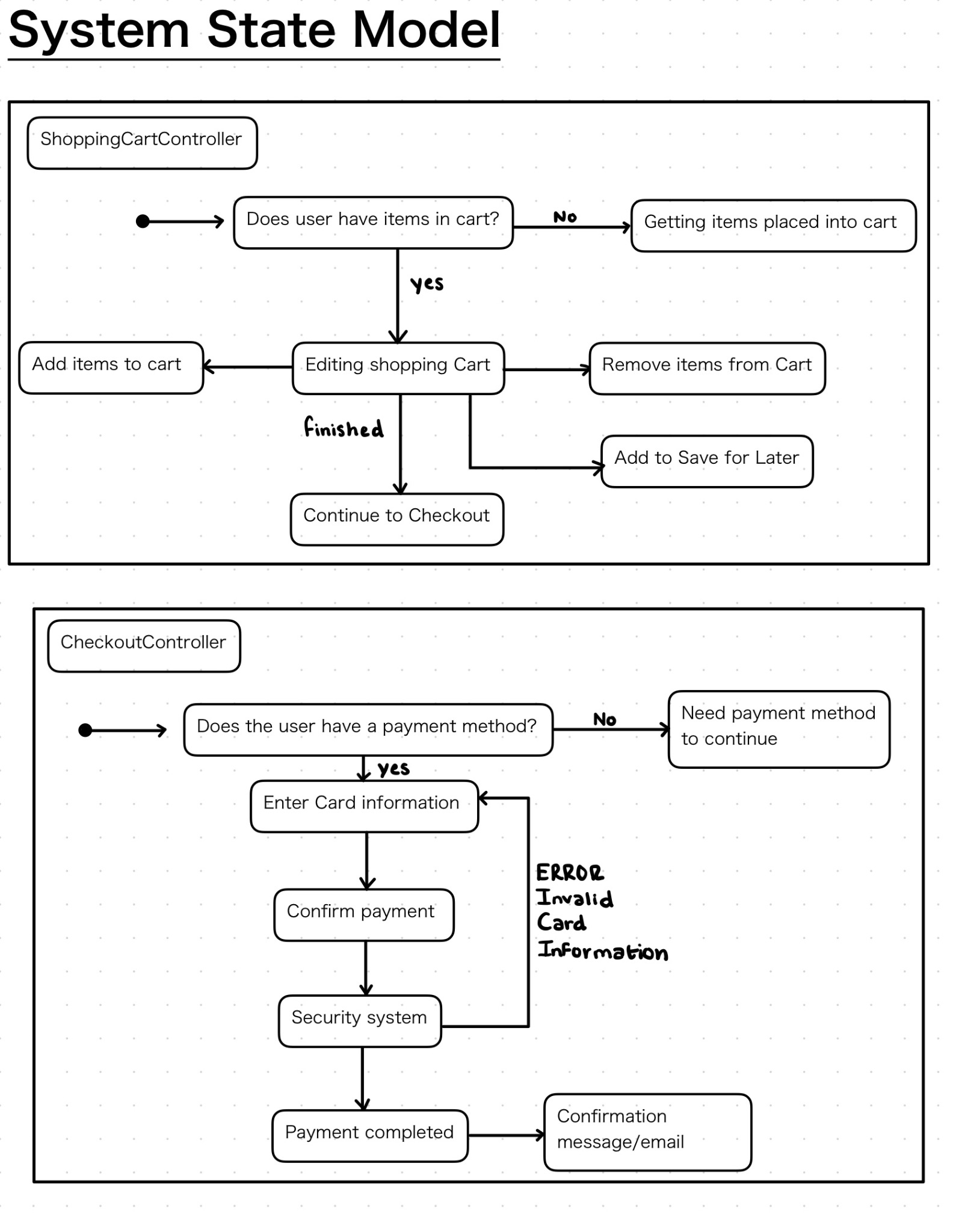
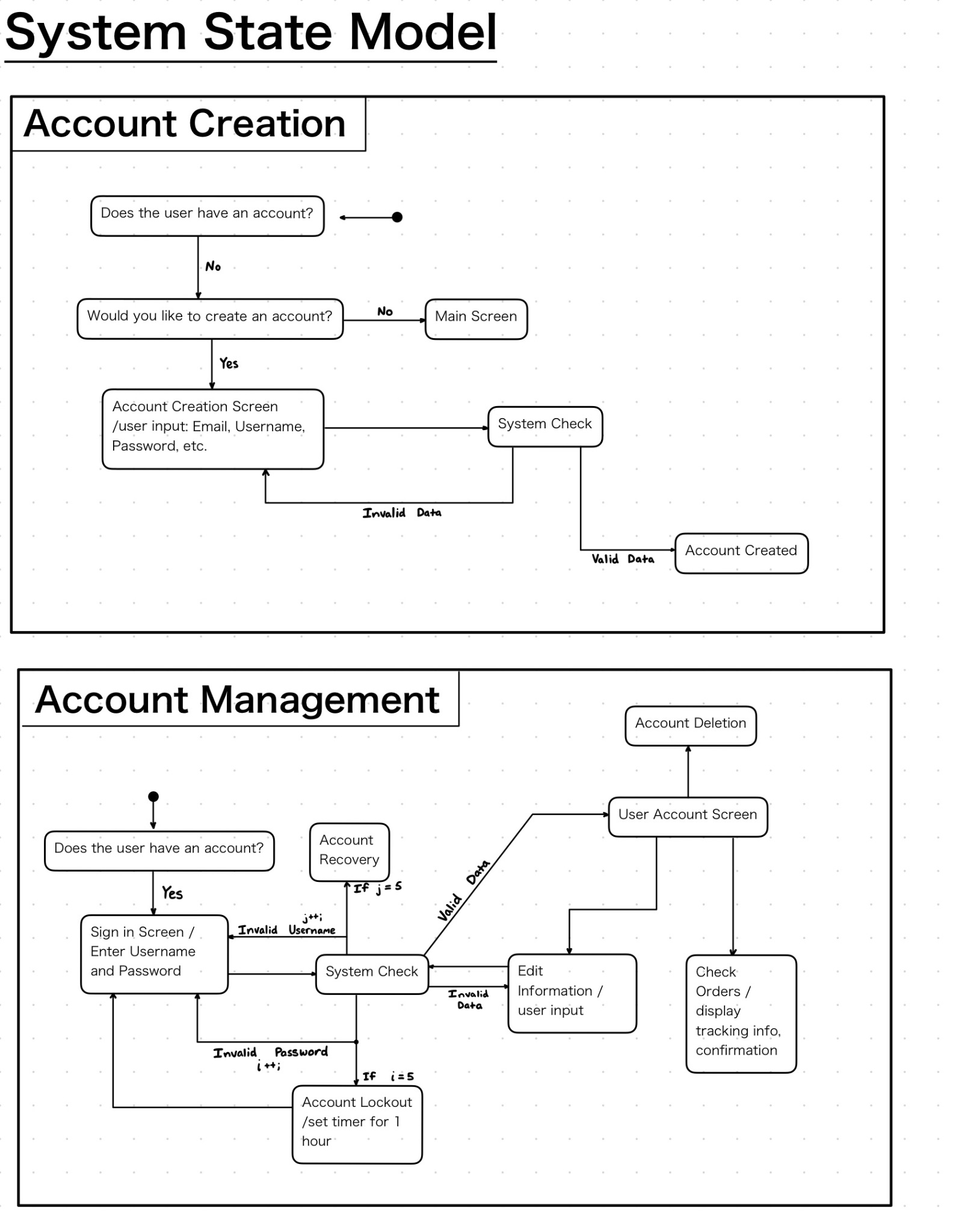
1. Account Creation: Does the user have an account? If not, then the user is prompted if they would like to create an account (This is not mandatory, as our ***System*** will work without an existing user). If they decide to, then they will be prompted to enter the following: first name, last name, email, phone number, address, a form of payment, username, and password. From here the ***System*** will check if the username and email address already exists.

2. Account Management: Does the user have an account? If so, then the user will proceed to their Account Management Event using their username and password. The ***System*** will validate the username, and password. There is the scenario if the user inputs their password wrong 5 times then, they’ll be locked out. Once the time limit has been reached, then they’ll be prompted to recover their account which is to create a new password. When the user successfully connects they can choose from three options, edit their existing information, check their orders, account deletion. If they choose account deletion the process is immediate. If they choose to see their orders, the page will display their existing orders, with tracking info and the confirmation number. Lastly, editing information will allow you to change: first name, last name, email, phone number, address, a form of payment, username, and password.

3. Managing Shopping Cart: Does the user have anything in their shopping cart? If not the user will remain in the managing shopping cart state. If a customer is looking for a specific category of item, it will have a filter for category and cost. Otherwise users will inherit the ability to make edits to their cart. Cart will also be updating in this state; as the customer is managing an account or checking out; this aspect allows the website to update its stock to display to the user how much stock is left or out of stock. This will also update the possible total cost of order every time an item gets removed or added. If out of stock then the item will remain in the shopping cart but the user cannot continue on to the checkout stage (Checkout button will be inactive). Users will be pointed in order to get to the “Checkout” stage then they will have to transfer the “out of stock” items into their “Save for later” list. Users also have the ability to add in items from their “Save for later” list. When adding an item into cart the customer has the ability to select the preferred quantity amount of that product and add it to their cart. Once the user is confident about their shopping cart and want to move to the “Checkout” step then they will click on the checkout button displayed on the website

4. Does the user have a payment method? If not then the user cannot proceed with checkout and remain in this state.. If a customer has a payment method, the user will be asked to enter name of card owner, credit card number, CVV, card expiration date, and billing address. Once customer supplies information a security method will then proceed to check confirmation on the card information . If the card information detects invalid payment method then the customer gets sent back to enter card information and displays an error message and states possible issues due to failure. If the security method then approves the card information for being valid then the customer will be sent to a page where they will be given a confirmation message displaying their name, address, payment method with numbers bleeped out, and a confirmation number with their expectancy date of arrival. The user will also be sent an email containing all this information as well and the customer has fully made a purchase.

5. Is the potential customer on our website? If so then, the GUI will need help to manage the flow of where the customer will want to go. This event is a little vague but as they decide what they want to do. It’ll condense into smaller events.



**State Descriptions**

State Name: Does the user have an account?

State Description: Prompt the user if they have an account with our website

Event Sequence that produces the state: From the main page of our website the user selected “Account”, which will take them to the sign in page.

Condition that characterizes this state: If the user has an account, they will either sign in or create an account.

Events accepted in this state: Account Creation, Account Management

State Name: Would you like to create an account?

State Description: Prompt the user if they would like to create an account.

Event Sequence that produces the state: From selecting the “Account button”, the user will decide whether they want to create one or not

Condition that characterizes this state: The user does not have an account.

Events accepted in this state: Account Creation

State Name: Main Screen

State Description: The user has either selected to not create an account, or just put our url in the search engine and was greeted to our main screen.

Event Sequence that produces the state: The user didn’t choose to create an account, and/or just searched our website.

Condition that characterizes this state: A directory that allow the user to navigate our website

Events accepted in this state: Account Creation, Session Controller

State Name: Account Creation Screen

State Description: In this menu/screen we will prompt the user to input their information such as email, username, password, etc; to create an account from it.

Event Sequence that produces the state: The user chose to create an account.

Condition that characterizes this state: The user does not have an account, and would like to create one

Events accepted in this state: Account Creation

State Name: System Check

State Description: With the users inputs we will validate the data they just put in to check for conflicts.

Event Sequence that produces the state: The user has input information that will need to be checked that it is valid.

Condition that characterizes this state: Depending where this event is located we will either prompt the user to reinput their data if it is incorrect, or if it is correct we will move them to the next stage. Also if a certain number of incorrect attempts is reached the user will be locked out from the system.

Events accepted in this state: Account Creation, Account Management

State Name: Account Created

State Description: The user has successfully input valid data and has created an account with our website.

Event Sequence that produces the state: The user has input information that is valid and that does not conflict.

Condition that characterizes this state: Their Information is input into our system so that duplicates aren’t created and safely protected

Events accepted in this state: Account Creation

State Name: Sign in Screen

State Description: The user wishes to sign in into their account

Event Sequence that produces the state: The user has an existing account.

Condition that characterizes this state: Their information shows up positive in our system that their account exists. The system will check that the password corresponds with the username, if either one of them is incorrect we increment a counter.

Events accepted in this state: Account Management

State Name: Account Recovery

State Description: The user forgot their login username and wishes to recover that.

Event Sequence that produces the state: The user has an existing account, and has failed to recall their username.

Condition that characterizes this state: If 5 incorrect username attempts are reached then they will be prompted to enter this stage.

Events accepted in this state: Account Management

State Name: Account Lockout

State Description: The user forgot their login password and will be punished for that.

Event Sequence that produces the state: The user has an existing account, and has failed to recall their password. Or someone is trying to break into the account.

Condition that characterizes this state: If 5 incorrect password attempts are reached then they will be locked out for an hour.

Events accepted in this state: Account Management

State Name: User Account Screen

State Description: The user has successfully logged into their account.

Event Sequence that produces the state: The user input a valid existing account and successfully logged in.

Condition that characterizes this state: The user has a valid account.

Events accepted in this state: Account Management

State Name: Edit Information

State Description: The user wishes to update their existing information.

Event Sequence that produces the state: The user has successfully logged in and has selected to “Edit Information”.

Condition that characterizes this state: If the data is sent to the system check to see if it is invalid or conflicting information. If the data is valid they’ll be returned to the User Account Screen.

Events accepted in this state: Account Management

State Name: Account Deletion

State Description: The user wishes to delete their account

Event Sequence that produces the state: The user successfully logged in, they could be unhappy with the website and wish to delete their information.

Condition that characterizes this state: remove() will be invoked and all data will be erased instantly.

Events accepted in this state: Account Management

State Name: Check Orders

State Description: The user wishes to view their existing orders.

Event Sequence that produces the state: The user successfully logged in, and wishes to view the status of their order, see the total cost, or what they ordered again.

Condition that characterizes this state: tracking information, order confirmation number, total price and items

Events accepted in this state: Account Management

State Name: Account

State Description: The user wishes to either create an account or manage their existing one.

Event Sequence that produces the state: From the main screen, the user selected “Account”.

Condition that characterizes this state: Account Management and Account Creation

Events accepted in this state: Session Controller

State Name: Account Management

State Description: The user wishes to manage their existing account.

Event Sequence that produces the state: The user wishes to manage their account and will have to sign in to get there.

Condition that characterizes this state: Edit Information, Account Deletion, Check Orders, Account Lockout/Recovery.

Events accepted in this state: Session Controller

State Name: Account Creation

State Description: The user wishes to manage their existing account.

Event Sequence that produces the state: The user wishes to manage their account and will have to sign in to get there.

Condition that characterizes this state: Edit Information, Account Deletion, Check Orders, Account Lockout/Recovery.

Events accepted in this state: Session Controller

State Name: Shopping Cart

State Description: The user wishes to manage their shopping cart.

Event Sequence that produces the state: From the main screen, the user selected “Shopping Cart”.

Condition that characterizes this state: Shopping Cart Manager, Checkout Process

Events accepted in this state: Session Controller

State Name: Shopping Cart Manager

State Description: The user wishes to manage their shopping cart.

Event Sequence that produces the state: From the main screen, the user selected “Shopping Cart”.

Condition that characterizes this state: Remove/Add Items, Check to see if it is in stock, sum of current items, save for later, and checkout.

Events accepted in this state: Session Controller

State Name: Checkout Process

State Description: The user wishes to exit and pay for the items in their shopping cart, to get them shipped.

Event Sequence that produces the state: There are items in their shopping cart and the total is displayed for them to pay for.

Condition that characterizes this state: If they are an existing user, then the information will be automatically populated accordingly. If they don’t then they will have input things such as Payment, Address, Email, etc.

Events accepted in this state: Session Controller

State Name: Browse

State Description: The user wishes to browse the catalog of items that our website is selling.

Event Sequence that produces the state: From the main screen, the user selected “Browse”.

Condition that characterizes this state: Look at items in detail, such as price, category, description, name, if it’s in stock.

Events accepted in this state: Session Controller





