Name: Nguyễn Hải Tiến Phát  
ID: 521H0126

**Câu 2:**

**Functional Requirements:**

1. The ticket vending machine shall provide a user interface to allow the user to select the desired mode of payment.
2. The ticket vending machine shall provide a menu display of potential destinations for the user to select from.
3. The ticket vending machine shall accept credit card payment and issue a paper ticket with a barcode.
4. The ticket vending machine shall accept digital wallet payment and display a QR code for the user to scan.
5. The ticket vending machine shall validate credit card transactions before issuing the ticket.
6. The ticket vending machine shall display the price of the ticket to the user before processing the payment.
7. The ticket vending machine shall dispense the purchased ticket to the user after payment confirmation.
8. The ticket vending machine shall provide a receipt to the user upon request.

**Non-functional Requirements:**

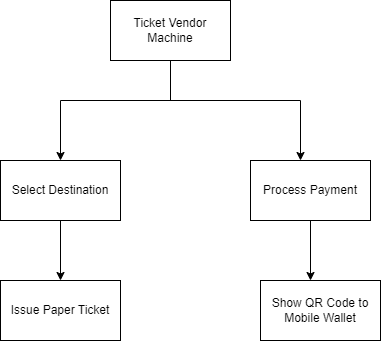
1. The ticket vending machine shall have a response time of no more than 5 seconds for user inputs.
2. The ticket vending machine shall have a reliability of 99.9% and should not fail more than once per month.
3. The ticket vending machine shall have a maximum downtime of 30 minutes for maintenance purposes.
4. The ticket vending machine shall have a user-friendly interface that is easy to understand and navigate.
5. The ticket vending machine shall have a maximum noise level of 60 decibels to avoid causing discomfort to the users.

**Domain Requirements:**

1. The ticket vending machine shall comply with all applicable laws and regulations.
2. The ticket vending machine shall be designed to be easily accessible for users with disabilities.
3. The ticket vending machine shall be equipped with a security system to prevent theft or vandalism.
4. The ticket vending machine shall be able to operate in different weather conditions.
5. The ticket vending machine shall be connected to a centralized system to ensure ticket synchronization across different machines.
6. The ticket vending machine shall be able to provide real-time data on ticket sales and usage to the system administrators.

**Câu 3:**

**Use case diagram:**



**Description:**

The Ticket Vendor Machine is a software system that sells public transportation tickets to users. There are two modes of payment available to users: credit card and mobile wallet.

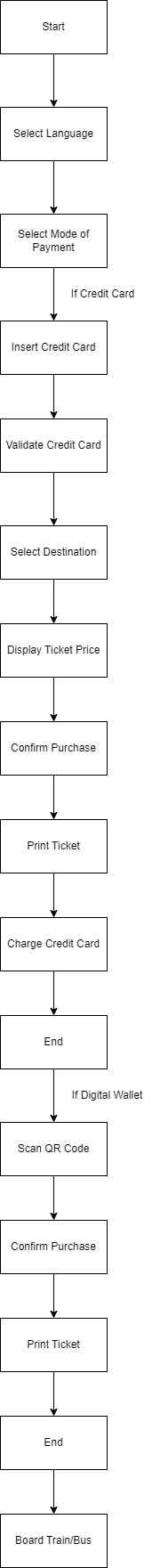
The use case diagram shows the different interactions between the user and the Ticket Vendor Machine system. There are four main use cases:

1. Select Destination: The user initiates this use case by pressing the start button on the ticket vendor machine. The system displays a menu of potential destinations for the user to select. The user selects their destination from the menu.
2. Process Payment: Once the user has selected their destination, they must pay for the ticket. The system offers two modes of payment: credit card and mobile wallet. If the user selects credit card, they are prompted to input their credit card information. The system then validates the credit card transaction and charges the user's account. If the user selects mobile wallet, the system generates a QR code for the user's mobile wallet to scan and pay for the ticket.
3. Issue Paper Ticket: If the user selects to pay with a credit card, the system issues a paper ticket with a barcode that the user can use to board public transportation. This use case involves generating a paper ticket with a barcode and initiating a credit card transaction.
4. Show QR Code to Mobile Wallet: If the user selects to pay with a mobile wallet, the system displays a QR code for the user's mobile wallet to scan and pay for the ticket. This use case involves generating a QR code and displaying it on the ticket vendor machine for the user's mobile wallet to scan.

Overall, the Ticket Vendor Machine system is designed to provide a simple and automated way for users to purchase public transportation tickets using their preferred mode of payment.

**Câu 4:**

**Activity Diagram:**



**Here are the steps involved:**

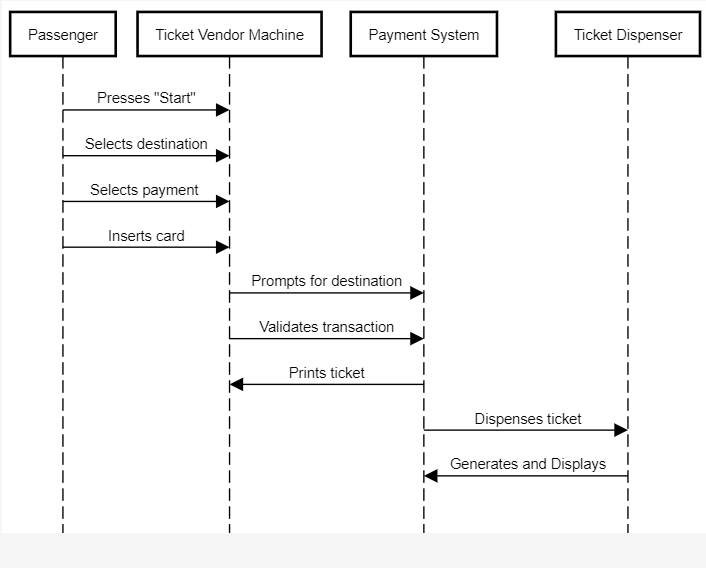
1. The passenger approaches the ticket vendor machine and selects their preferred language from the available options.
2. The machine displays a welcome message and prompts the passenger to select their mode of payment (credit card or digital wallet).
3. If the passenger selects credit card, the machine prompts them to insert their credit card into the card reader.
4. The machine validates the credit card and prompts the passenger to select their destination from a list of available options.
5. The passenger selects their destination, and the machine displays the ticket price.
6. The passenger confirms the ticket purchase, and the machine prints a paper ticket with a barcode.
7. The machine charges the passenger's credit card for the ticket price.
8. If the passenger selects digital wallet, the machine prompts them to scan the QR code displayed on the screen with their mobile device.
9. The passenger's digital wallet app opens, and they confirm the ticket purchase.
10. The machine displays a confirmation message and prints a paper ticket with a barcode.
11. The passenger boards the bus/train and presents their paper ticket or mobile device with the QR code for validation.

**Explain more from this Activity Diagram**

Break down the activity diagram step by step to explain each part in more detail:

1. Start: This is the starting point of the process. The passenger approaches the ticket vendor machine.
2. Select Language: The machine displays a menu of available languages, and the passenger selects their preferred language.
3. Select Mode of Payment: The machine prompts the passenger to select their mode of payment - either credit card or digital wallet.
4. If Credit Card: If the passenger selects credit card as their mode of payment, the machine prompts them to insert their credit card into the card reader.
5. Insert Credit Card: The passenger inserts their credit card into the machine.
6. Validate Credit Card: The machine validates the credit card to ensure that it is valid and has sufficient funds.
7. Select Destination: The machine displays a list of available destinations, and the passenger selects their desired destination.
8. Display Ticket Price: The machine displays the ticket price for the selected destination.
9. Confirm Purchase: The passenger confirms that they want to purchase a ticket.
10. Print Ticket: The machine prints a paper ticket with a barcode.
11. Charge Credit Card: The machine charges the passenger's credit card for the ticket price.
12. End: The process is complete.
13. If Digital Wallet: If the passenger selects digital wallet as their mode of payment, the machine displays a QR code that the passenger can scan with their mobile device.
14. Scan QR Code: The passenger scans the QR code displayed on the machine with their mobile device.
15. Confirm Purchase: The passenger confirms that they want to purchase a ticket using their digital wallet app.
16. Print Ticket: The machine prints a paper ticket with a barcode.
17. End: The process is complete.
18. Board Train/Bus: The passenger presents their paper ticket or mobile device with the QR code to the ticket inspector for validation before boarding the train/bus.

**Câu 5:**

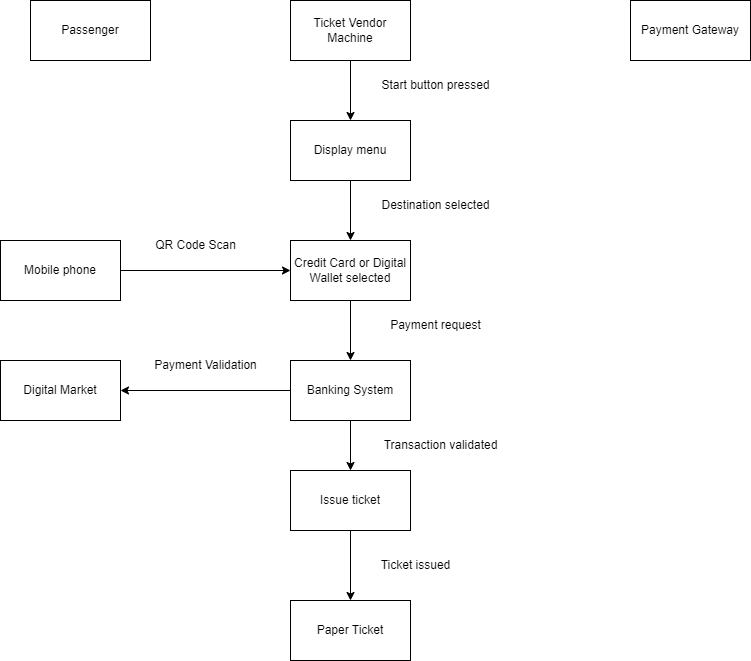


The steps involved in the "Buy a ticket" use case scenario for the Ticket Vendor Machine:

1. The passenger presses the "Start" button on the Ticket Vendor Machine.
2. The Ticket Vendor Machine displays a menu of potential destinations for the passenger to select from.
3. The passenger selects a destination.
4. The Ticket Vendor Machine prompts the passenger to choose a mode of payment (e.g. credit card, digital wallet).
5. If the passenger selects credit card, the Ticket Vendor Machine prompts them to insert their card.
6. The Ticket Vendor Machine validates the credit card transaction.
7. The Ticket Vendor Machine prints out a paper ticket with a barcode and dispenses it to the passenger.
8. If the passenger selects digital wallet, the Ticket Vendor Machine displays a QR code for the passenger to scan with their mobile device.
9. The passenger scans the QR code with their mobile device and completes the transaction.
10. The Ticket Vendor Machine generates a digital ticket and displays it for the passenger to verify.

These steps describe the process that a passenger goes through when they use the Ticket Vendor Machine to buy a ticket for public transportation.

**+ Communication Diagram:**



The steps involved in the use case are as follows:

1. The Passenger presses the Start button on the Ticketing Vendor Machine.
2. The Ticketing Vendor Machine displays a menu of potential destinations and prompts the Passenger to select a destination.
3. The Passenger selects a destination.
4. If the Passenger chooses to pay with a credit card, the Ticketing Vendor Machine prompts the Passenger to insert their credit card.
5. The Payment Gateway validates the credit transaction.
6. If the credit transaction is valid, the Ticketing Vendor Machine issues a paper ticket with a barcode and charges the Passenger's credit card account.
7. If the Passenger chooses to pay with a digital wallet, the Ticketing Vendor Machine displays a QR Code.
8. The Passenger scans the QR Code using their mobile phone and makes a payment through their Digital Wallet System.
9. The Digital Wallet System validates the payment.
10. If the payment is valid, the Ticketing Vendor Machine issues a paper ticket with a barcode.

+ Steps involved in the "Buy a ticket" use case for the Ticketing Vendor Machine, explained in more detail:

1. The Passenger presses the Start button on the Ticketing Vendor Machine. This is the trigger that starts the use case.
2. The Ticketing Vendor Machine displays a menu of potential destinations and prompts the Passenger to select a destination. The Passenger can use the machine's interface to make their selection.
3. The Passenger selects a destination. Once the Passenger has selected a destination, the use case proceeds differently depending on the chosen payment method:

* If the Passenger chooses to pay with a credit card, the Ticketing Vendor Machine prompts the Passenger to insert their credit card. The Passenger inserts their credit card into the machine's card reader.
* If the Passenger chooses to pay with a digital wallet, the Ticketing Vendor Machine displays a QR code. The Passenger scans the QR code with their mobile phone to initiate the payment process.

1. The Payment Gateway validates the credit transaction or the Digital Wallet System validates the payment. The Payment Gateway or Digital Wallet System communicates with the Banking System to verify that the Passenger's account has sufficient funds.
2. If the credit transaction or payment is valid, the Ticketing Vendor Machine issues a paper ticket with a barcode and charges the Passenger's credit card account or acknowledges the payment. The Passenger takes the paper ticket and keeps it with them until they reach their destination.
3. The use case is complete and the Ticketing Vendor Machine returns to its initial state, ready for the next Passenger to use.

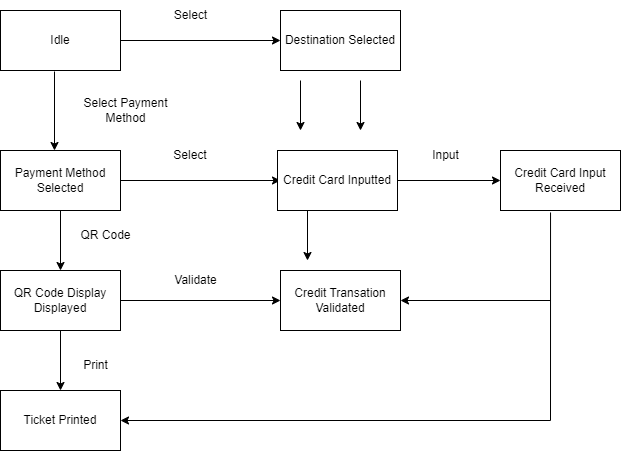
+ The Payment Gateway is an essential component of the Smart Ticketing System. It handles the processing of credit card transactions made by the Passenger.

Here's how it works:

1. Once the Passenger selects their destination and chooses to pay with a credit card, the Ticketing Vendor Machine prompts the Passenger to insert their credit card into the machine's card reader.
2. The Ticketing Vendor Machine then sends the credit card information securely to the Payment Gateway for processing. This information typically includes the card number, expiration date, and security code.
3. The Payment Gateway then communicates with the Passenger's bank to verify that their account has sufficient funds to cover the cost of the ticket.
4. If the bank confirms that the transaction is valid, the Payment Gateway sends an approval message back to the Ticketing Vendor Machine.
5. The Ticketing Vendor Machine then prints out the paper ticket with a barcode and charges the Passenger's credit card account for the cost of the ticket.
6. If the bank denies the transaction, the Payment Gateway sends a decline message back to the Ticketing Vendor Machine, and the Passenger will need to use a different payment method or card to purchase their ticket.

Overall, the Payment Gateway plays a crucial role in securely processing credit card transactions and ensuring that the Passenger's funds are transferred correctly.

**+ State chart Diagram:**



+ The initial state, or starting point, of the Ticket Vendor Machine is the Idle state. When the machine is first turned on or reset, it will be in this state, waiting for the user to interact with it by selecting a destination or payment method.

In the Idle state, the machine is not processing any transactions and is waiting for the user to initiate an event by selecting a destination or payment method. Once an event occurs, such as the user selecting a destination or payment method, the machine transitions to the appropriate state and begins processing the transaction.

+ When we talk about the possible events in a state chart diagram, we are identifying the different actions that can trigger a transition from one state to another. Each event represents a different interaction between the user and the system.

Here are the possible events that can occur in the Ticket Vendor Machine:

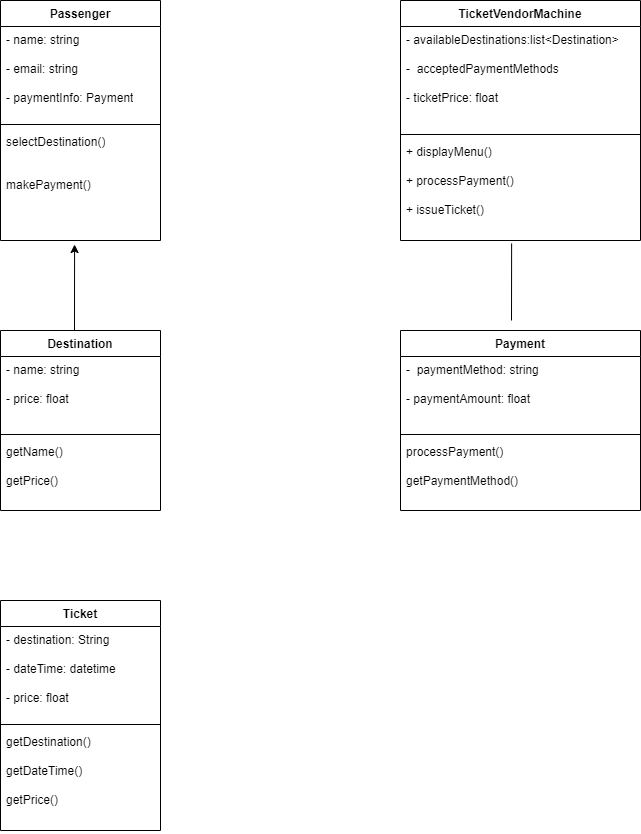
1. Select Destination: This event occurs when the user selects a destination from the menu display. It triggers a transition from the Idle state to the Destination Selected state.
2. Select Payment Method: After the user selects a destination, this event occurs when the user selects a payment method. It triggers a transition from the Destination Selected state to the Payment Method Selected state.
3. Input Credit Card: This event occurs when the user inputs their credit card information. It triggers a transition from the Payment Method Selected state to the Credit Card Input Received state.
4. Validate Credit Transaction: After the user inputs their credit card information, this event occurs when the system verifies the credit card information and validates the credit transaction. It triggers a transition from the Credit Card Input Received state to the Credit Transaction Validated state.
5. Display QR Code: If the user selects to pay using a digital wallet, this event occurs when the system displays a QR code that the user can scan with their mobile device to complete the payment. It triggers a transition from the Payment Method Selected state to the QR Code Displayed state.

+ When we talk about the possible states in a state chart diagram, we are identifying the different states that the system can be in based on the events that occur. Each state represents a different configuration of the system, with its own set of rules and behaviors.

Here is a breakdown of each possible state for the Ticket Vendor Machine:

1. Idle: This is the initial state of the Ticket Vendor Machine, where it is waiting for the user to initiate an action.
2. Destination selected: This state is reached when the user selects their destination from the list of potential destinations displayed on the machine's menu.
3. Payment method selected: After the user selects their destination, they are prompted to select their preferred mode of payment. This state is reached when the user selects their preferred payment method, such as credit card or digital wallet.
4. Credit card inputted: If the user selects to pay with a credit card, they are prompted to input their credit card details. This state is reached when the user inputs their credit card information and submits it for validation.
5. Credit card input received: If the user selects to pay with a credit card, they are prompted to input their credit card details. This state is reached when the user inputs their credit card information.
6. Credit transaction validated: Once the user submits their credit card information, the system validates the transaction by verifying that the card information is correct and that the user has sufficient funds to purchase the ticket. If the transaction is validated, the system moves to this state.
7. QR code displayed: If the user selects to pay with a digital wallet, the system generates a QR code that the user can scan with their mobile device to complete the payment. This state is reached when the QR code is displayed on the machine's screen.
8. Ticket printed: Finally, after the payment has been completed, the system prints the ticket with a barcode that the user can use to enter the public transportation system. This is the final state of the Ticket Vendor Machine.

**+ Class Diagram:**



**Step 1: Identify the main actors involved in the use case**

In software development, we use the concept of "actors" to represent the different people or systems that interact with a given system. In this use case, there are two main actors involved:

* Passenger: This is the person who wants to buy a ticket. They will interact with the Ticket Vendor Machine to select their destination, choose their payment method, and receive their ticket.
* Ticket Vendor Machine: This is the system that enables the passenger to buy a ticket. It will provide a menu of available destinations, process the passenger's payment information, and issue a ticket.

By identifying these two actors, we can better understand the different types of interactions that will take place between them and the system. This helps us to design the system in a way that meets the needs of all stakeholders involved.

**Step 2: Identify the main functionalities of the Ticket Vendor Machine**

* Display menu of potential destinations: The Ticket Vendor Machine will display a list of available destinations that the passenger can choose from. This menu should be easy to navigate and clearly indicate the different options available to the passenger.
* Allow passenger to select a destination: Once the menu of potential destinations is displayed, the passenger should be able to select the destination they want to travel to. This selection could be made by pressing a button or touching a screen, for example.
* Process payment: The Ticket Vendor Machine should allow the passenger to pay for their ticket using one of the accepted payment methods. These may include credit card, digital wallet, or other forms of payment. The system should securely process the payment information and ensure that the correct amount is charged to the passenger.
* Issue a ticket: Once the payment has been processed and validated, the Ticket Vendor Machine should issue a ticket to the passenger. This ticket may be in the form of a paper ticket with a barcode or a QR code that can be scanned by transportation staff to verify that the passenger has paid for their journey. The ticket should clearly indicate the destination, date and time of travel, and any other relevant information.

Overall, the main goal of the Ticket Vendor Machine is to enable passengers to easily purchase tickets for public transportation using a variety of payment methods. By providing a clear and user-friendly interface, processing payments securely, and issuing tickets efficiently, the system can help to streamline the process of buying tickets and improve the overall passenger experience.

**Step 3: Identify the main classes involved in the use case**

* Passenger: This class represents the passenger who wants to buy a ticket. It may include attributes such as the passenger's name, payment information, and selected destination.
* TicketVendorMachine: This class represents the system that enables the passenger to buy a ticket. It may include methods for displaying the menu of potential destinations, processing payment information, and issuing a ticket.
* Destination: This class represents the possible destinations that the passenger can select from. It may include attributes such as the name of the destination and the price of a ticket to that destination.
* Payment: This class represents the payment process, which may involve credit card or digital wallet payment. It may include methods for securely processing payment information and ensuring that the correct amount is charged to the passenger.
* Ticket: This class represents the ticket that is issued to the passenger. It may include attributes such as the destination, date and time of travel, and any other relevant information.

Overall, these classes work together to enable the Ticket Vendor Machine to function properly and provide passengers with a seamless ticket-buying experience. By breaking down the system into smaller, more manageable classes, we can better understand how the different components of the system interact and ensure that the system is designed in a way that meets the needs of all stakeholders involved.

**Step 4: Define the relationships between the classes**

* Passenger has a relationship with TicketVendorMachine: This relationship represents the fact that the passenger interacts with the TicketVendorMachine in order to buy a ticket. The Passenger class uses the methods provided by the TicketVendorMachine class to select a destination, process payment, and obtain a ticket.
* TicketVendorMachine has a relationship with Destination: This relationship represents the fact that the TicketVendorMachine displays a menu of potential destinations to the passenger and allows them to select one. The Destination class provides the information necessary to display the list of available destinations and allow the passenger to make a selection.
* TicketVendorMachine has a relationship with Payment: This relationship represents the fact that the TicketVendorMachine processes the payment information provided by the passenger. The Payment class provides the methods necessary to securely process the payment information and ensure that the correct amount is charged to the passenger.
* TicketVendorMachine has a relationship with Ticket: This relationship represents the fact that the TicketVendorMachine issues a ticket to the passenger once payment has been processed. The Ticket class provides the information necessary to create a ticket with the correct destination, date and time of travel, and any other relevant information.

Overall, these relationships help to define the interactions between the different classes and ensure that the system works as intended. By clearly defining the relationships between the classes, we can better understand how the different components of the system fit together and ensure that the system is designed in a way that meets the needs of all stakeholders involved.

**Step 5: Define the attributes and methods of each class**

**Passenger:**

+ Attributes:

* name: the name of the passenger
* email: the email address of the passenger
* payment information: the payment information provided by the passenger (such as credit card details or digital wallet information)

+ Methods:

* selectDestination(): allows the passenger to select a destination from the available options
* makePayment(): initiates the payment process and provides the necessary payment information to the TicketVendorMachine class

**TicketVendorMachine:**

+ Attributes:

* list of available destinations: a list of the available destinations for the passenger to choose from
* payment methods accepted: the payment methods that are accepted by the system (such as credit card or digital wallet)
* current ticket price: the current price of the ticket, which may vary depending on the destination and other factors

+ Methods:

* displayMenu(): displays the list of available destinations to the passenger
* processPayment(): processes the payment information provided by the passenger and confirms the payment has been made
* issueTicket(): creates and issues a ticket to the passenger once payment has been confirmed

**Destination:**

+ Attributes:

* name: the name of the destination
* price: the price of the ticket to the destination

+ Methods:

* getName(): returns the name of the destination
* getPrice(): returns the price of the ticket to the destination

**Payment:**

+ Attributes:

* payment method: the method of payment (such as credit card or digital wallet)
* payment amount: the amount that needs to be paid for the ticket

+ Methods:

* processPayment(): securely processes the payment information provided by the passenger and confirms the payment has been made
* getPaymentMethod(): returns the payment method used by the passenger

**Ticket:**

+ Attributes:

* destination: the destination of the ticket
* date and time of the trip: the date and time that the ticket is valid for
* ticket price: the price of the ticket

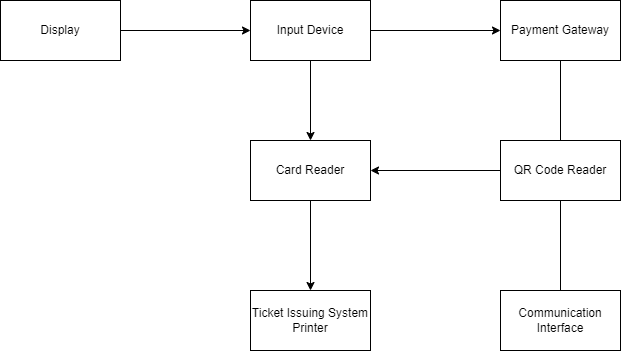
+ Methods:

* getDestination(): returns the destination of the ticket
* getDateTime(): returns the date and time that the ticket is valid for
* getPrice(): returns the price of the ticket

Overall, these attributes and methods help to define the behavior and functionality of each class within the system. By clearly defining these attributes and methods, we can ensure that each class is designed in a way that meets the needs of the system and the users involved.

**Câu 7:**

**+ Architecture Design:**



Step 1: Identify the components of the system

The first step is to identify the different components of the Ticket Vendor Machine. Based on the requirements, we can identify the following components:

1. Display: This component is responsible for displaying the menu of potential destinations and other messages to the user.

Explanation of the component responsible for displaying the menu of potential destinations and other messages to the user:

1. The component receives input from the user interface or application logic, such as a request to display a menu of potential destinations.
2. The component processes the input, which may involve retrieving relevant data from a database or other source, formatting the data into a user-friendly format, and generating any necessary visual elements such as buttons or icons.
3. The component then displays the menu or message to the user, typically on a screen or other visual display. This may involve rendering text, images, or other multimedia elements, and may use various techniques such as scrolling, animation, or highlighting to draw the user's attention to important information.
4. The component may also provide interactive functionality, such as allowing the user to select a destination from the menu, or to navigate through additional pages or menus to find more information.
5. Finally, the component may handle any user interactions or feedback related to the displayed information, such as acknowledging a user's selection or displaying an error message if the user input is invalid.

Overall, the display component plays a critical role in presenting information to the user in a clear and intuitive way, and in facilitating user interactions with the system.

1. Input Device: This component allows the user to select the destination and input their credit card or digital wallet information.

Explanation of the input device component:

1. The component receives input from the user, such as a request to select a destination or to input payment information.
2. The input device may take various forms, such as a keyboard, touchscreen, mouse, or voice recognition system, depending on the specific implementation.
3. The user interacts with the input device to provide the necessary input, such as selecting a destination from a menu or entering their credit card or digital wallet information.
4. The input device then sends the user's input to the application logic or other relevant components of the system.
5. Depending on the specific implementation, the input device may also provide feedback to the user, such as displaying the text they have entered or highlighting the selected destination.
6. The input device may also perform additional processing on the user input, such as verifying the validity of a credit card number or checking the user's input against a list of available destinations.
7. Finally, the input device may handle any errors or exceptions related to the user input, such as displaying an error message if the user's credit card information is invalid.

Overall, the input device component plays a critical role in allowing the user to interact with the system and input the necessary information to complete their transaction. It must be designed with the user's needs and capabilities in mind, and must also be able to handle a wide range of input types and formats.

1. Payment Gateway: This component is responsible for handling the credit card or digital wallet transaction.

Explanation of the payment gateway component:

1. The component receives the user's payment information, such as their credit card number or digital wallet account details.
2. The payment gateway verifies the validity of the payment information, such as checking that the credit card number is correctly formatted and that the account has sufficient funds.
3. If the payment information is valid, the payment gateway sends the payment request to the relevant payment processor or financial institution.
4. The payment processor or financial institution then processes the payment request, which may involve additional verification steps such as checking the user's identity or performing fraud detection.
5. Once the payment has been authorized and processed, the payment processor or financial institution sends a response back to the payment gateway indicating the status of the transaction, such as "approved" or "declined".
6. Based on the response from the payment processor or financial institution, the payment gateway updates the status of the transaction in the system and sends a confirmation message to the user indicating whether the payment was successful or not.
7. In addition to handling payment transactions, the payment gateway may also perform additional functions such as managing user accounts, storing transaction history, or generating reports and analytics.

Overall, the payment gateway component is a critical part of any payment processing system, as it handles the complex interactions with payment processors and financial institutions, ensuring the security and reliability of payment transactions. It must be designed to handle a wide range of payment methods and formats, as well as to comply with relevant industry standards and regulations.

1. Ticket Issuing System: This component is responsible for issuing the paper ticket with a barcode or displaying a QR code for mobile payment.

Explanation of the ticket issuing system component:

1. Once the payment gateway confirms that the user's payment has been successful, the ticket issuing system component receives the payment confirmation.
2. The ticket issuing system generates a unique identifier or code for the ticket, such as a barcode or QR code, which can be scanned to validate the ticket.
3. Depending on the specific implementation, the ticket issuing system may then either print a paper ticket with the barcode or display the QR code on a screen for mobile payment.
4. If a paper ticket is printed, the ticket issuing system sends the necessary instructions to a printer connected to the system.
5. Once the ticket has been issued, the ticket issuing system updates the status of the ticket in the system, indicating that it has been issued and is now valid for use.
6. When the user presents the ticket for validation, the barcode or QR code is scanned by a ticket validation system, which confirms that the ticket is valid and has not been used before.
7. If the ticket is valid, the user is granted access to the relevant service or destination, such as boarding a train or entering a theme park.
8. Depending on the specific implementation, the ticket issuing system may also perform additional functions such as tracking usage statistics, generating reports and analytics, or handling refunds and cancellations.

Overall, the ticket issuing system component plays a critical role in ensuring that users have a valid ticket to access the relevant service or destination. It must be designed to handle a range of ticket formats and payment methods, as well as to ensure the security and integrity of the ticketing system.

1. Communication Interface: This component is responsible for communicating with the backend system to validate the transaction and update the database.

Explanation of the communication interface component:

1. When the user completes a transaction, such as purchasing a ticket or making a reservation, the communication interface component receives the relevant transaction data, such as the user's payment details and the service or destination requested.
2. The communication interface component then communicates with the backend system, which may be a remote server or database, to validate the transaction and update the relevant records.
3. The communication interface component sends the transaction data to the backend system using a standard communication protocol, such as HTTP or TCP/IP.
4. The backend system then processes the transaction data, performing any necessary validation checks and updating the relevant records in the database.
5. Once the transaction has been processed, the backend system sends a response back to the communication interface component indicating whether the transaction was successful or not.
6. Based on the response from the backend system, the communication interface component updates the status of the transaction in the system and sends a confirmation message to the user indicating whether the transaction was successful or not.
7. Depending on the specific implementation, the communication interface component may also perform additional functions such as logging transaction data, generating reports and analytics, or handling errors and exceptions.

Overall, the communication interface component plays a critical role in ensuring that the transaction data is properly transmitted and processed by the backend system. It must be designed to handle a range of communication protocols and formats, as well as to ensure the security and reliability of the transaction processing system.

Step 2: Draw the architecture diagram

Once we have identified the components, we can draw the architecture diagram. The diagram shows the main components and their interactions. The arrows indicate the flow of information between the components.

The diagram shows that the user interacts with the Input Device to select the destination and input their payment information. The Payment Gateway handles the payment transaction, and the Ticket Issuing System issues the paper ticket with a barcode or displays a QR code for mobile payment. The Communication Interface is responsible for communicating with the backend system to validate the transaction and update the database.

Step 3: Add more details

Depending on the specific requirements and the technology used, we may need to add more details to the architecture diagram. For example, we may need to include components such as the Card Reader, QR Code Reader, Printer, and Cash Handling System.

The Card Reader reads the credit card information, and the QR Code Reader reads the QR code from the user's mobile phone. The Printer prints the paper ticket with a barcode. The Cash Handling System handles cash payments.

+ Explanation of how the Card Reader and QR Code Reader components work in the context of the Ticket Vendor Machine:

1. When a user selects the option to pay with a credit card, they are prompted to insert their credit card into the Card Reader component of the machine.
2. The Card Reader reads the information from the credit card's magnetic strip or chip using a card reader head. This information includes the cardholder's name, credit card number, expiration date, and security code.
3. The Ticket Vendor Machine then sends this information to the Payment Gateway component of the system for processing.
4. The Payment Gateway contacts the bank that issued the credit card to validate the transaction and ensure that the card has sufficient funds for the transaction.
5. If the transaction is validated, the Payment Gateway sends a message back to the Ticket Vendor Machine to indicate that the payment was successful.
6. The Ticket Vendor Machine then prints the paper ticket with a barcode, which the user can use to access public transportation.

+ How the QR Code Reader works when a user selects the option to pay with a digital wallet:

1. When a user selects the option to pay with a digital wallet, they are prompted to scan the QR code displayed on the machine's screen using their mobile phone's camera.
2. The QR Code Reader component of the machine reads the QR code and decodes it into a digital format that can be processed by the system.
3. The Ticket Vendor Machine then sends this information to the Payment Gateway component of the system for processing.
4. The Payment Gateway contacts the user's digital wallet provider to validate the transaction and ensure that the user has sufficient funds for the transaction.
5. If the transaction is validated, the Payment Gateway sends a message back to the Ticket Vendor Machine to indicate that the payment was successful.
6. The Ticket Vendor Machine then displays a QR code on the screen or prints the paper ticket with a barcode, which the user can use to access public transportation.

Overall, the Card Reader and QR Code Reader components of the Ticket Vendor Machine enable users to pay for their public transportation tickets using a credit card or a digital wallet. These components play an important role in the overall functionality of the machine and help to ensure a seamless user experience for passengers.

+ Explanation of how the Printer and Cash Handling System components work in the context of the Ticket Vendor Machine:

1. When a user selects the option to pay with a credit card or digital wallet, the Ticket Vendor Machine processes the payment and generates a ticket with a barcode.
2. The Printer component of the machine prints the paper ticket with the barcode on it.
3. The user can then take the paper ticket and use it to access public transportation.

+ How the Cash Handling System works when a user selects the option to pay with cash:

1. When a user selects the option to pay with cash, they are prompted to insert the cash payment into the Cash Handling System component of the machine.
2. The Cash Handling System component accepts the cash payment and validates it to ensure that it is genuine currency.
3. Once the payment is validated, the Ticket Vendor Machine generates a ticket with a barcode.
4. The Printer component of the machine prints the paper ticket with the barcode on it.
5. The user can then take the paper ticket and use it to access public transportation.

Overall, the Printer and Cash Handling System components of the Ticket Vendor Machine enable users to pay for their public transportation tickets using either a credit card or digital wallet or cash. These components play an important role in the overall functionality of the machine and help to ensure a seamless user experience for passengers.

+ Explanation of how the overall architecture of the Ticket Vendor Machine works in the context of the Smart Ticketing System:

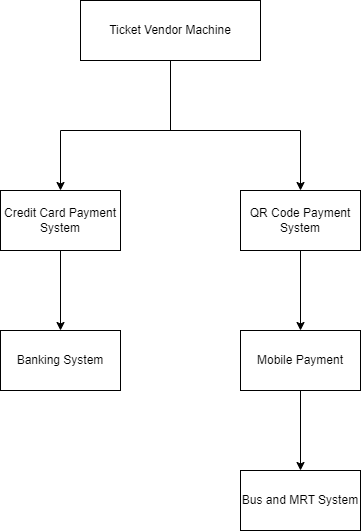
1. A user approaches the Ticket Vendor Machine and selects the mode of payment they want to use (credit card, digital wallet, or cash).
2. If the user selects the credit card or digital wallet payment options, the Card Reader or QR Code Reader component of the machine is activated, respectively.
3. The user provides their credit card information or scans the QR code on their mobile device.
4. The Ticket Vendor Machine sends the payment information to the Payment Gateway component of the Smart Ticketing System.
5. The Payment Gateway contacts the user's bank or digital wallet provider to validate the payment and ensure that there are sufficient funds available.
6. If the payment is validated, the Payment Gateway sends a message back to the Ticket Vendor Machine to indicate that the payment was successful.
7. The Ticket Vendor Machine generates a ticket with a barcode using the Printer component and dispenses the ticket to the user.
8. If the user selects the cash payment option, they insert the payment into the Cash Handling System component of the machine.
9. The Cash Handling System accepts and validates the payment.
10. The Ticket Vendor Machine generates a ticket with a barcode using the Printer component and dispenses the ticket to the user.
11. The user takes the ticket and uses it to access public transportation.

Overall, the Ticket Vendor Machine is a critical component of the Smart Ticketing System, enabling users to purchase tickets using various payment methods and providing a seamless experience for passengers. The machine relies on several key components, including the Card Reader, QR Code Reader, Payment Gateway, Cash Handling System, and Printer, to process payments and dispense tickets to users.

**Deployment Diagram:**

+ How to draw a deployment diagram for the Ticket Vendor Machine:

1. Start by drawing a rectangle in the center of the page to represent the Ticket Vendor Machine.
2. Draw two arrows pointing to the Ticket Vendor Machine from the top and bottom, representing the connection to the network.
3. Draw two rectangles above the Ticket Vendor Machine, representing the Credit Card Payment System and the QR Code Payment System.
4. Draw arrows connecting the Credit Card Payment System and the QR Code Payment System to the Ticket Vendor Machine.
5. Draw a rectangle on the right-hand side of the page to represent the Banking System.
6. Draw an arrow connecting the Credit Card Payment System to the Banking System.
7. Draw two rectangles on the left-hand side of the page to represent the Bus and MRT Systems.
8. Draw arrows connecting the Bus and MRT Systems to the Ticket Vendor Machine.
9. Label the arrows and rectangles with the appropriate names to complete the deployment diagram.



+ Explanation of the deployment diagram for the Ticket Vendor Machine:

1. The Ticket Vendor Machine is a hardware device that serves as an automated ticket-issuing system for public transportation such as buses and MRTs.

Explanation of this statement:

1. The Ticket Vendor Machine is a physical device, which means it is a hardware device. It is designed specifically to issue tickets for public transportation such as buses and MRTs.
2. The Ticket Vendor Machine is automated, which means it is designed to operate without human intervention. Passengers can use the machine to purchase tickets without the need for a ticketing agent or conductor.
3. The primary function of the Ticket Vendor Machine is to issue tickets for public transportation. It allows passengers to select their desired destination and mode of payment, and then issues a ticket based on that information.
4. The Ticket Vendor Machine is designed specifically for public transportation such as buses and MRTs. This means it is optimized for the unique requirements of these modes of transportation, such as the need to issue tickets quickly and efficiently, and to handle large volumes of passengers.

Overall, the statement describes the key features and functions of the Ticket Vendor Machine, and highlights its importance in providing a convenient and efficient way for passengers to purchase tickets for public transportation.

1. The Ticket Vendor Machine is connected to a network that enables it to communicate with other systems and devices.

Explanation of this statement:

1. The Ticket Vendor Machine is designed to communicate with other systems and devices. This means that it is connected to a network that enables it to send and receive data.
2. By being connected to a network, the Ticket Vendor Machine is able to access data and services that are provided by other systems and devices. This includes systems such as the Credit Card Payment System and the QR Code Payment System, which enable passengers to make payments for their tickets.
3. The Ticket Vendor Machine may also be connected to other systems and devices that are used in the transportation network. For example, it may be connected to the Bus and MRT Systems, which provide information about routes and schedules.
4. The network connection also allows the Ticket Vendor Machine to send data to other systems and devices. For example, it may send information about the number of tickets sold to a central monitoring system.
5. The network connection provides a way for the Ticket Vendor Machine to be remotely monitored and managed. This means that system administrators can check the status of the machine, troubleshoot any issues, and update the software as needed.

Overall, the statement describes how the Ticket Vendor Machine is designed to be part of a larger networked system, which enables it to communicate with other systems and devices, access data and services, and be remotely managed and monitored.

1. The Credit Card Payment System is a separate system that allows passengers to make payments for their tickets using their credit card. When a passenger selects this mode of payment, the Ticket Vendor Machine communicates with the Credit Card Payment System to process the payment and issue the ticket.

Explanation of this statement:

1. The Credit Card Payment System is a separate system that is designed specifically to process payments made using credit cards.
2. When a passenger selects the credit card payment option on the Ticket Vendor Machine, the machine communicates with the Credit Card Payment System to initiate the payment process.
3. The Ticket Vendor Machine sends the payment request to the Credit Card Payment System, which processes the payment using the passenger's credit card information.
4. Once the payment has been processed, the Credit Card Payment System sends a confirmation message to the Ticket Vendor Machine.
5. Based on the confirmation message, the Ticket Vendor Machine then issues the ticket to the passenger. The ticket may be a paper ticket with a barcode, which is printed by the machine.
6. The Ticket Vendor Machine may also update its internal records to reflect the sale of the ticket and the amount paid by the passenger.

Overall, the statement describes how the Credit Card Payment System works in conjunction with the Ticket Vendor Machine to enable passengers to make payments for their tickets using their credit card. The Ticket Vendor Machine communicates with the Credit Card Payment System to initiate the payment process, and then issues the ticket to the passenger once the payment has been processed.

1. The QR Code Payment System is another system that allows passengers to make payments using a digital wallet and scanning a QR code. When a passenger selects this mode of payment, the Ticket Vendor Machine generates a QR code that the passenger scans using their mobile phone. The Ticket Vendor Machine then communicates with the QR Code Payment System to process the payment and issue the ticket.

Explanation:

1. The QR Code Payment System is a separate system that is designed specifically to process payments made using digital wallets and QR codes.
2. When a passenger selects the QR Code payment option on the Ticket Vendor Machine, the machine generates a QR code that the passenger scans using their mobile phone.
3. The Ticket Vendor Machine sends the payment request to the QR Code Payment System, which processes the payment using the passenger's digital wallet information and the QR code data.
4. Once the payment has been processed, the QR Code Payment System sends a confirmation message to the Ticket Vendor Machine.
5. Based on the confirmation message, the Ticket Vendor Machine then issues the ticket to the passenger. The ticket may be a paper ticket with a barcode, which is printed by the machine.
6. The Ticket Vendor Machine may also update its internal records to reflect the sale of the ticket and the amount paid by the passenger.

Overall, the statement describes how the QR Code Payment System works in conjunction with the Ticket Vendor Machine to enable passengers to make payments for their tickets using a digital wallet and a QR code. The Ticket Vendor Machine generates the QR code that the passenger scans using their mobile phone, and then communicates with the QR Code Payment System to initiate the payment process. Once the payment has been processed, the Ticket Vendor Machine issues the ticket to the passenger.

1. The Banking System is another separate system that the Credit Card Payment System communicates with to validate the credit card transaction and charge the passenger's credit card account.

Explanation:

1. The Banking System is a separate system that is used by the Credit Card Payment System to validate credit card transactions and charge the passenger's credit card account.
2. When a passenger selects the credit card payment option on the Ticket Vendor Machine, the machine sends the payment request to the Credit Card Payment System.
3. The Credit Card Payment System then communicates with the Banking System to validate the transaction and ensure that the passenger has sufficient funds to pay for the ticket.
4. If the transaction is validated and the passenger has sufficient funds, the Banking System charges the passenger's credit card account.
5. Once the payment has been processed, the Credit Card Payment System sends a confirmation message to the Ticket Vendor Machine.
6. Based on the confirmation message, the Ticket Vendor Machine then issues the ticket to the passenger. The ticket may be a paper ticket with a barcode, which is printed by the machine.
7. The Ticket Vendor Machine may also update its internal records to reflect the sale of the ticket and the amount paid by the passenger.

Overall, the statement describes how the Credit Card Payment System works with the Banking System to validate credit card transactions and charge the passenger's account. The Ticket Vendor Machine communicates with the Credit Card Payment System to initiate the payment process, and the Credit Card Payment System communicates with the Banking System to validate the transaction and charge the passenger's account. Once the payment has been processed, the Ticket Vendor Machine issues the ticket to the passenger.

The Bus and MRT Systems are separate systems that the Ticket Vendor Machine communicates with to determine the available routes and prices for the passenger's desired destination. The Ticket Vendor Machine then issues the appropriate ticket.

Explanation:

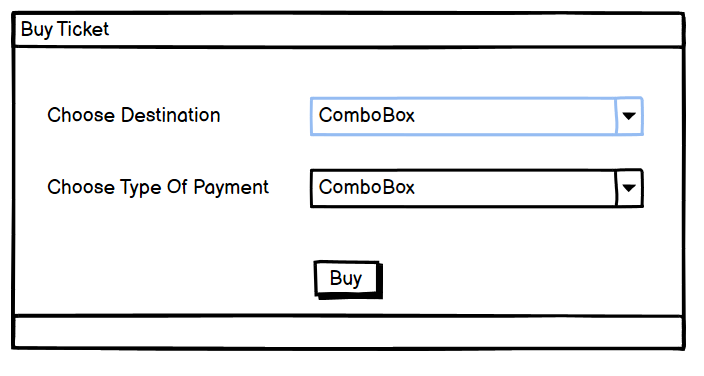
1. The Bus and MRT Systems are separate systems that the Ticket Vendor Machine communicates with to obtain information about the available routes and prices for the passenger's desired destination.
2. When a passenger selects their desired destination on the Ticket Vendor Machine, the machine sends a request to the Bus and MRT Systems for information about the available routes and prices.
3. The Bus and MRT Systems receive the request and use their own databases to determine the available routes and prices.
4. The Bus and MRT Systems then send the available route and price information back to the Ticket Vendor Machine.
5. The Ticket Vendor Machine receives the information and selects the appropriate ticket based on the passenger's desired destination and payment method.
6. If the passenger has selected the credit card payment option, the Ticket Vendor Machine will generate a paper ticket with a barcode and send a payment request to the Credit Card Payment System.
7. If the passenger has selected the QR code payment option, the Ticket Vendor Machine will generate a QR code for the passenger to scan with their mobile phone and send a payment request to the QR Code Payment System.
8. Once the payment has been processed and validated, the Ticket Vendor Machine issues the appropriate ticket to the passenger.

Overall, the statement describes how the Ticket Vendor Machine communicates with the Bus and MRT Systems to obtain information about the available routes and prices for the passenger's desired destination. The Ticket Vendor Machine then selects the appropriate ticket based on the passenger's payment method and issues the ticket once payment has been processed and validated.

=>Overall, the deployment diagram shows how the Ticket Vendor Machine is integrated with other systems and devices to provide passengers with a convenient and efficient way of purchasing tickets for public transportation.

**Câu 6:**

MockUp



A mockup is a visual representation of a design or concept, typically created to provide an early preview or prototype of a product or system. It can be a hand-drawn sketch, a digital drawing, or even a full-fledged interactive prototype. In the case of your example, a mockup could be a visual design of a ticket purchasing interface that includes the following elements:

1. A "Buy Ticket" button or call-to-action that prompts the user to start the purchasing process.
2. A "Choose Destination" dropdown menu or ComboBox that allows the user to select their desired travel destination from a list of available options.
3. A "Choose Type of Payment" dropdown menu or ComboBox that enables the user to select their preferred payment method, such as credit card, PayPal, or bank transfer.
4. An "Buy" button that the user clicks to finalize their purchase and proceed with payment.

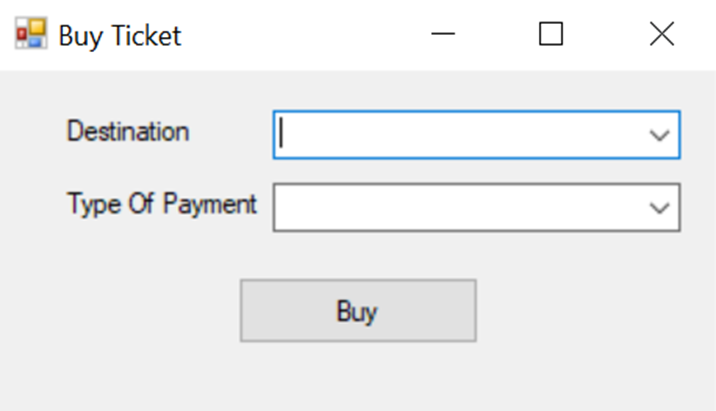
Here's how the mockup would work step-by-step:

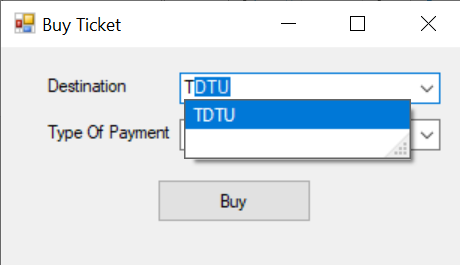
1. The user would first click the "Buy Ticket" button to initiate the purchase process.
2. The "Choose Destination" dropdown menu would appear, displaying a list of available destinations that the user can select from.
3. Once the user selects their desired destination from the dropdown menu, the "Choose Type of Payment" dropdown menu would appear, allowing them to select their preferred payment method.
4. After the user selects their payment method, they would click the "Buy" button to finalize their purchase.

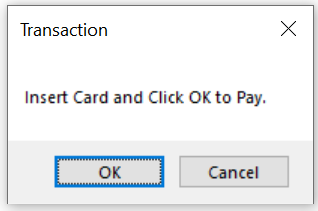
By creating a mockup of the ticket purchasing interface, designers can test and refine the user experience, identify potential issues or areas for improvement, and ensure that the final product meets the needs and expectations of its users.

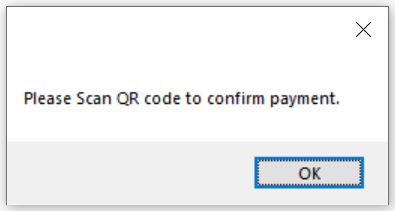
**Câu 8:**

Demo Explain











Step-by-step explanation of the process for purchasing a bus ticket using the demo explanation.

1. Choose Destination: The first step is to select the destination you want to travel to. You can do this by choosing from a list of available destinations or typing in the destination you want. The form/app will suggest the destination if it is available.
2. Choose Type of Payment: The next step is to select the type of payment you want to use. You can choose to pay by credit card or by digital wallet.
3. Click Buy: Once you have chosen your destination and payment method, you need to click the "Buy" button to start the transaction.
4. Credit Card Payment: If you choose to pay by credit card, the app will prompt you to insert your card and click "OK" to confirm the transaction. If you click "OK", the app will message you that the payment has been completed successfully. If you click "Cancel", you will receive a message saying that the purchase has been canceled and you will return to the previous form.
5. Digital Wallet Payment: If you choose to pay by digital wallet, the app will message you to scan and confirm. After clicking "OK", you will be redirected to a new form that displays a QR code.
6. Scan QR Code: You need to scan the QR code displayed on the new form. Once you have scanned the QR code, you need to click "OK" to confirm the payment and take the ticket. If you click "Cancel", you will return to the previous form and receive a message saying that the purchase has been canceled.
7. Ticket: After completing the transaction, you will receive a ticket that you can use to board the bus.

That is all about the Demo Explaination. The detail of code also have an explaination in it, if you need please read in the Form1.cs and Form2.cs.