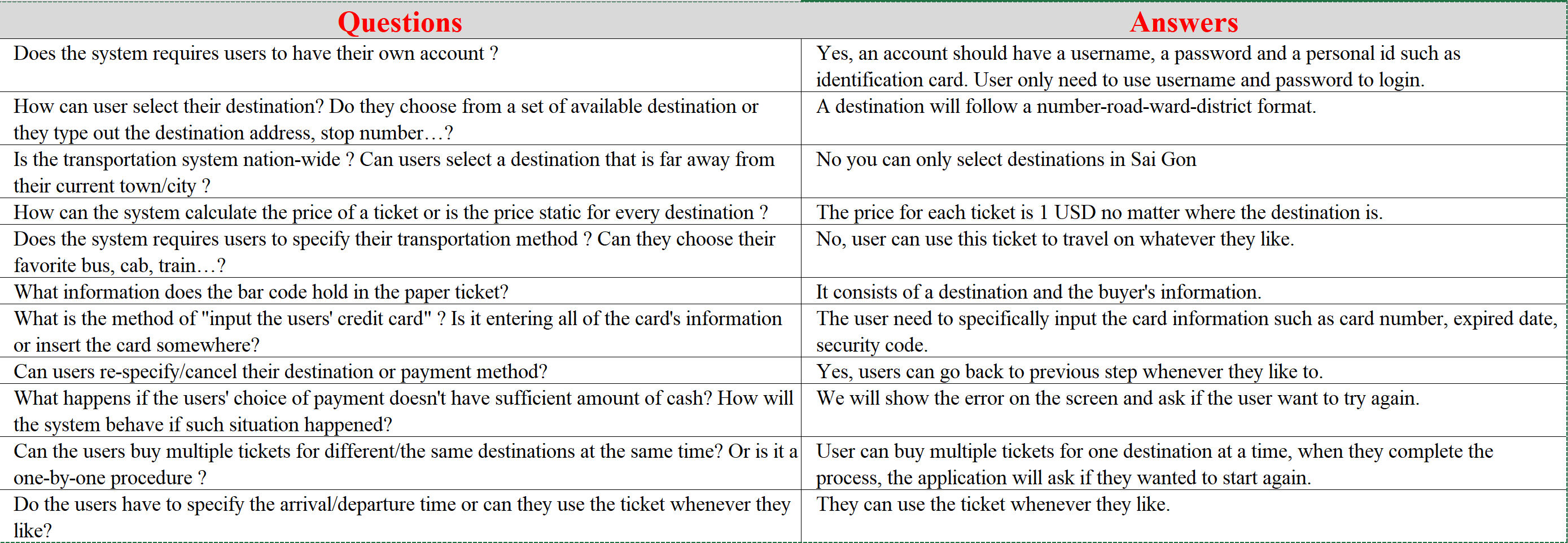
**Name: Nguyễn Phương Tài**

**ID: 521H0480**

1. Ask question to make a clear for above requirements and write them in the form of Excel (Q&A).



1. Write a set of functional, non-functional and domain requirements for a Ticket Vendor Machine. You can conduct this exercise to Excel or Word. Remember to concentrate on expectations of reliability and response time.

FUNCTIONAL REQUIREMENTS

* The system must allow users to register their account by using their username and password and personal ID.
* The system must allow users to login into their account also by their username and password.
* The system must allow users to select their destination by field-specific options. The choice goes by the order from City to District to Ward to Road.
* The system must allow users to select the number of ticket.
* The system must allow users to choose between credit card and online payment.
* The system must print a paper ticket for the users with the bar code.
* The system must allow users to abort the proccess if they wish to.

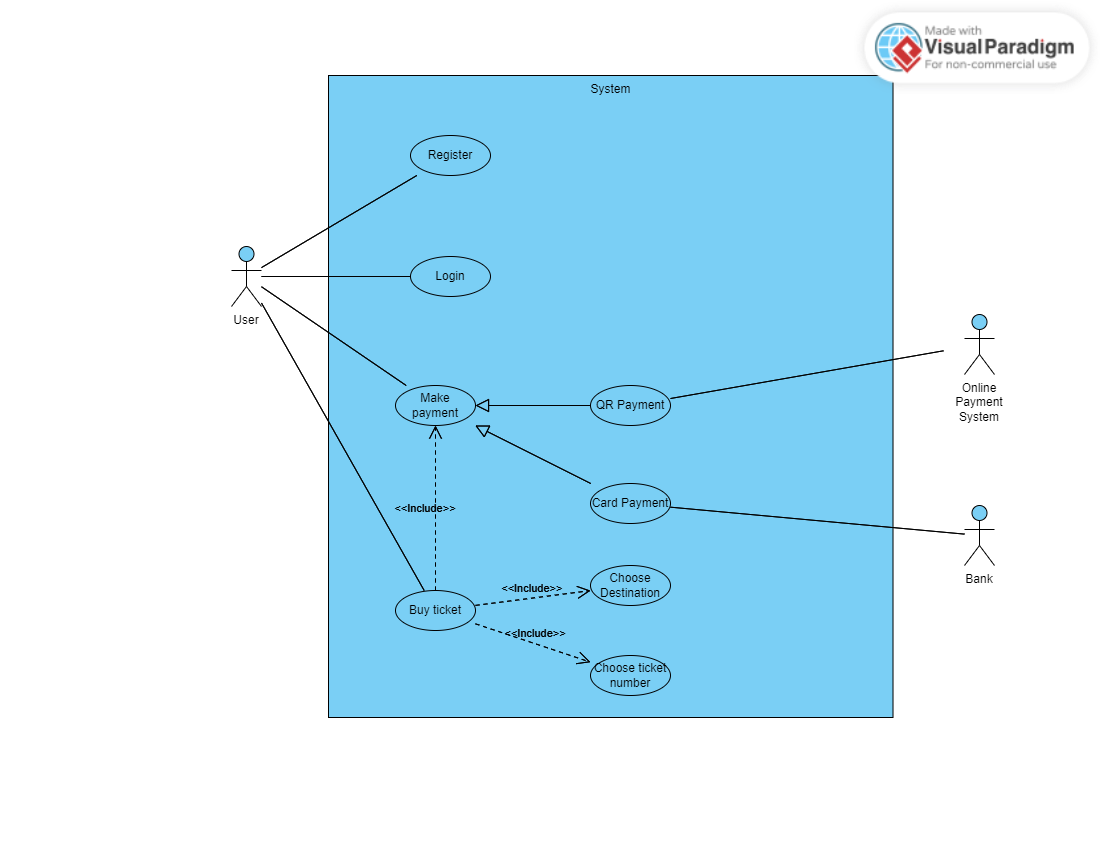
NON-FUNCTIONAL REQUIREMENTS

* The maximum time for views to load is 0.5 seconds.
* The acceptable rate of failure must be about a ratio of 1:100000.
* The maximum accepted percent of failure per day is 1%.
* The ticket must be issued no more than 5 seconds after validating the payment.
* The system must run 24/7
* The system must handle effieciently 10000 users at a time.

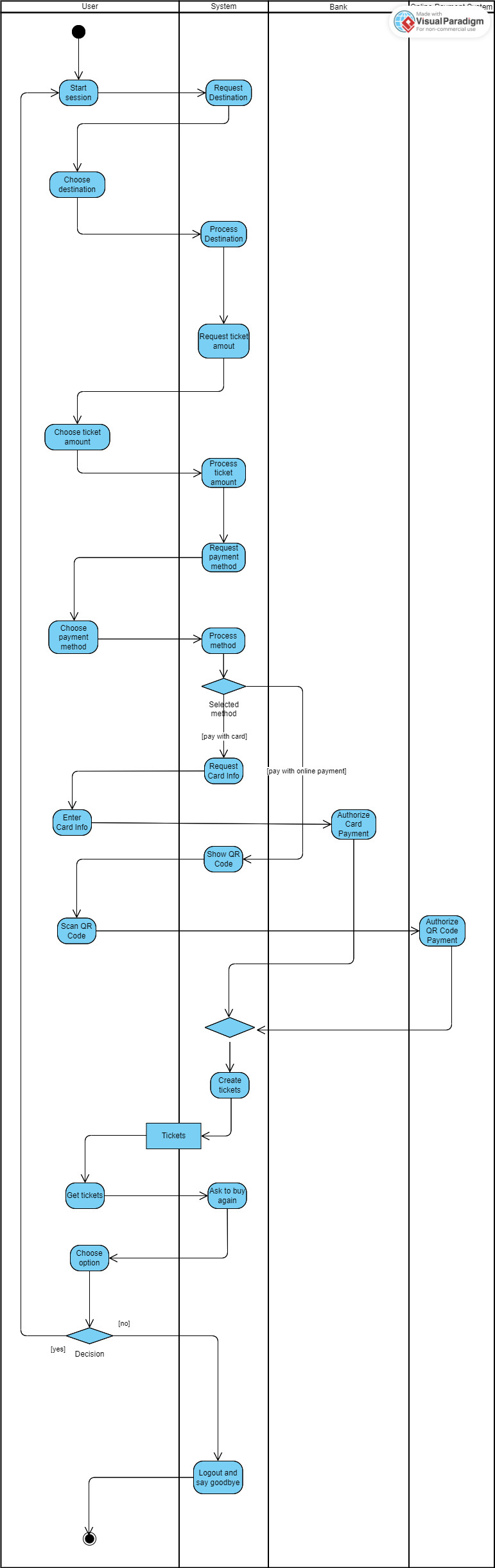
DOMAIN REQUIREMENTS

* Developers must be familiar with the ticket-vending process.
* All personel should develop the system according to the transport standard.

1. Develop Use Case modelling for Ticket Vendor Machine, you are also encouraged to make Use Case Description for each use case on your use case diagram.

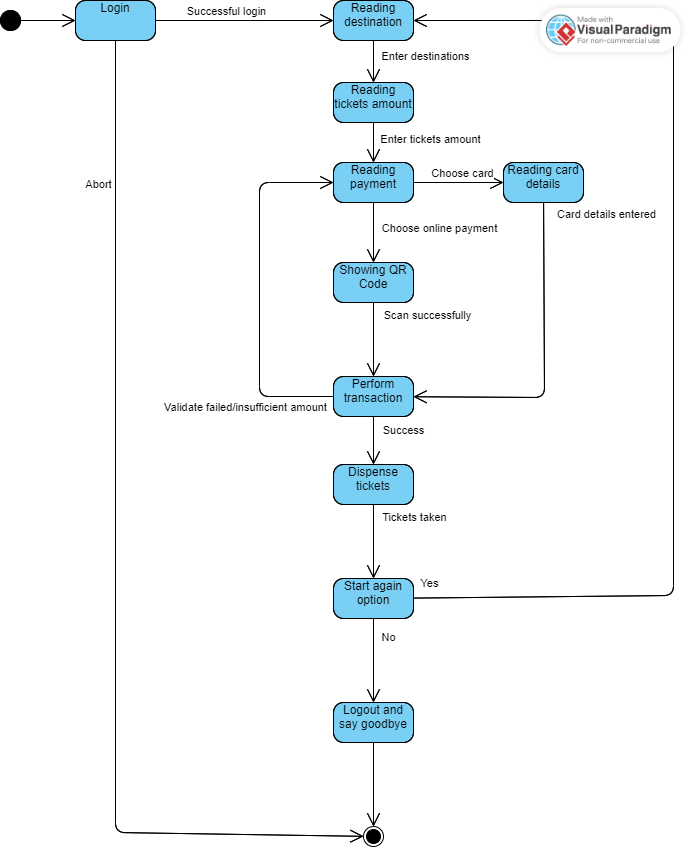


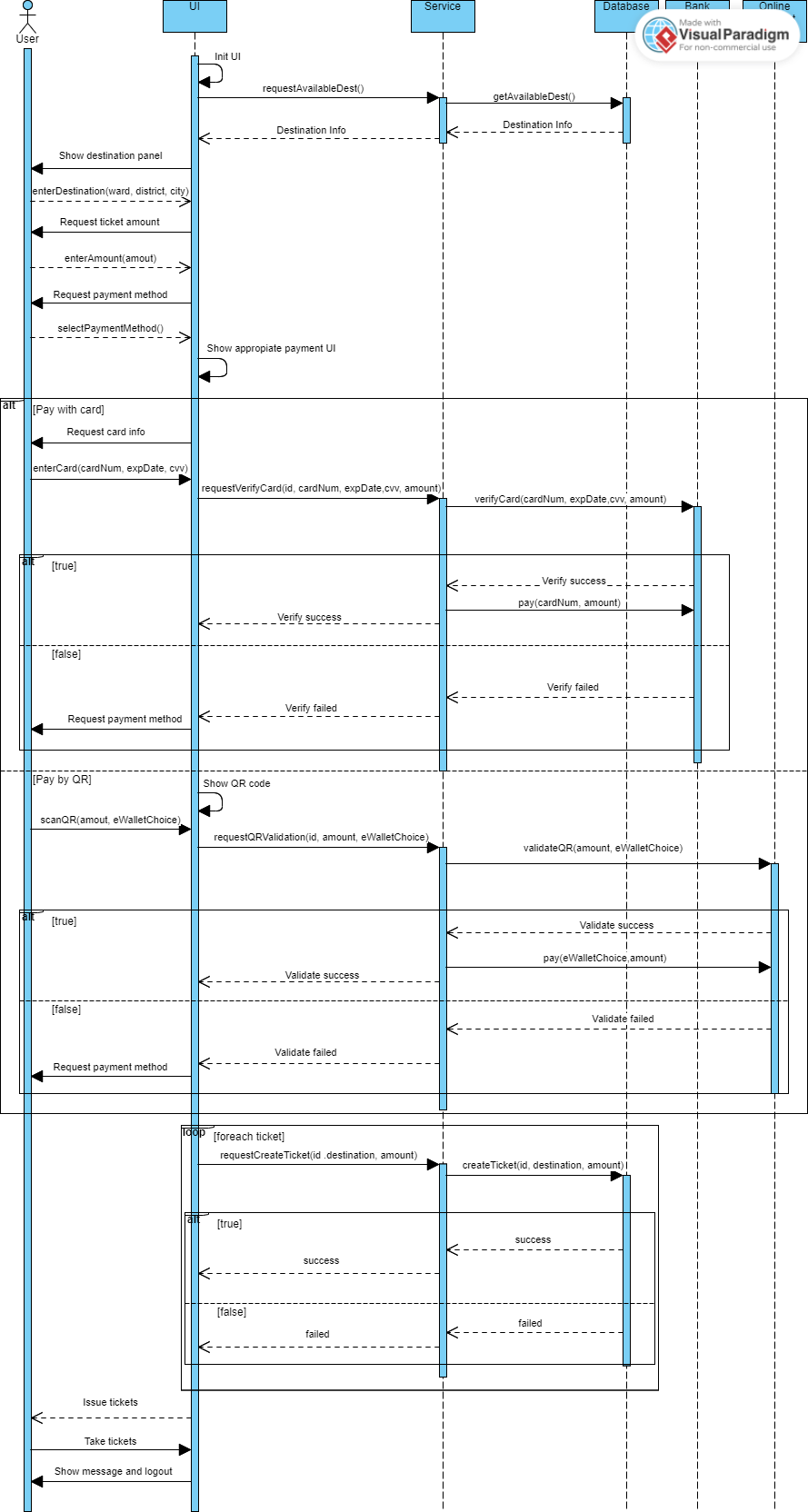
1. Make an Activity diagram to present the process of passenger’s buying a ticket from ticket vendor machine (Look like ATM) and the activity diagram for communication among systems if your system is integrated with other system like Momo, VNPay, ZaloPay,...etc

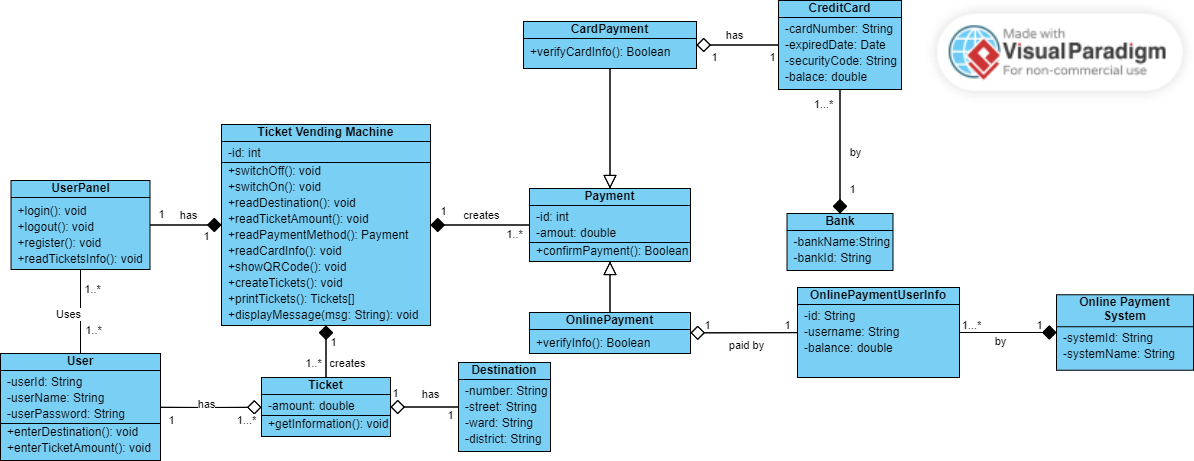


1. Let’s say that the Ticketing Vendor Machine have main use case: Buy a ticket then you are required to complete the sequence diagram, State chart diagram, and Class diagram.

* State Chart Diagram

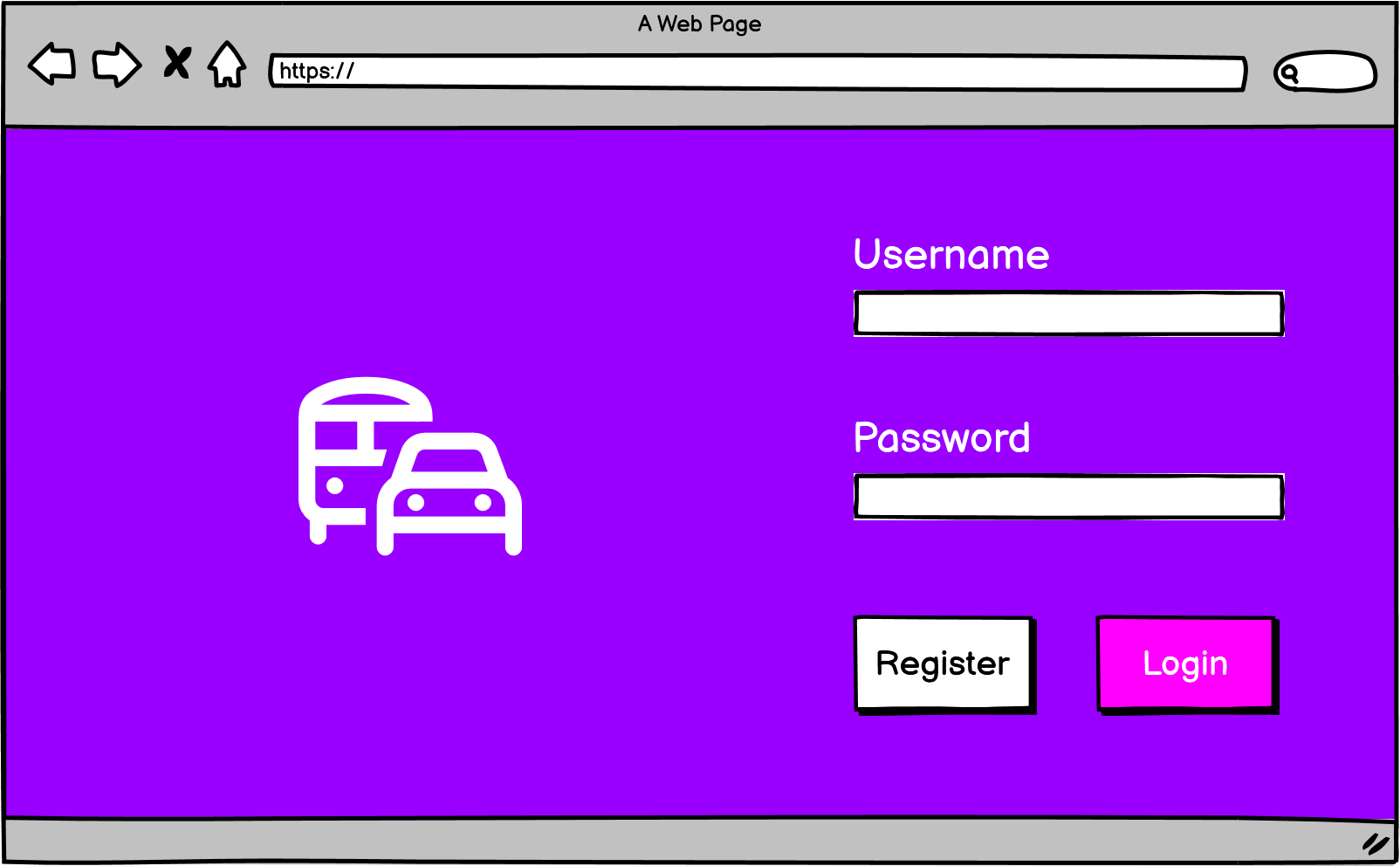


* Sequence diagram
* Class diagram

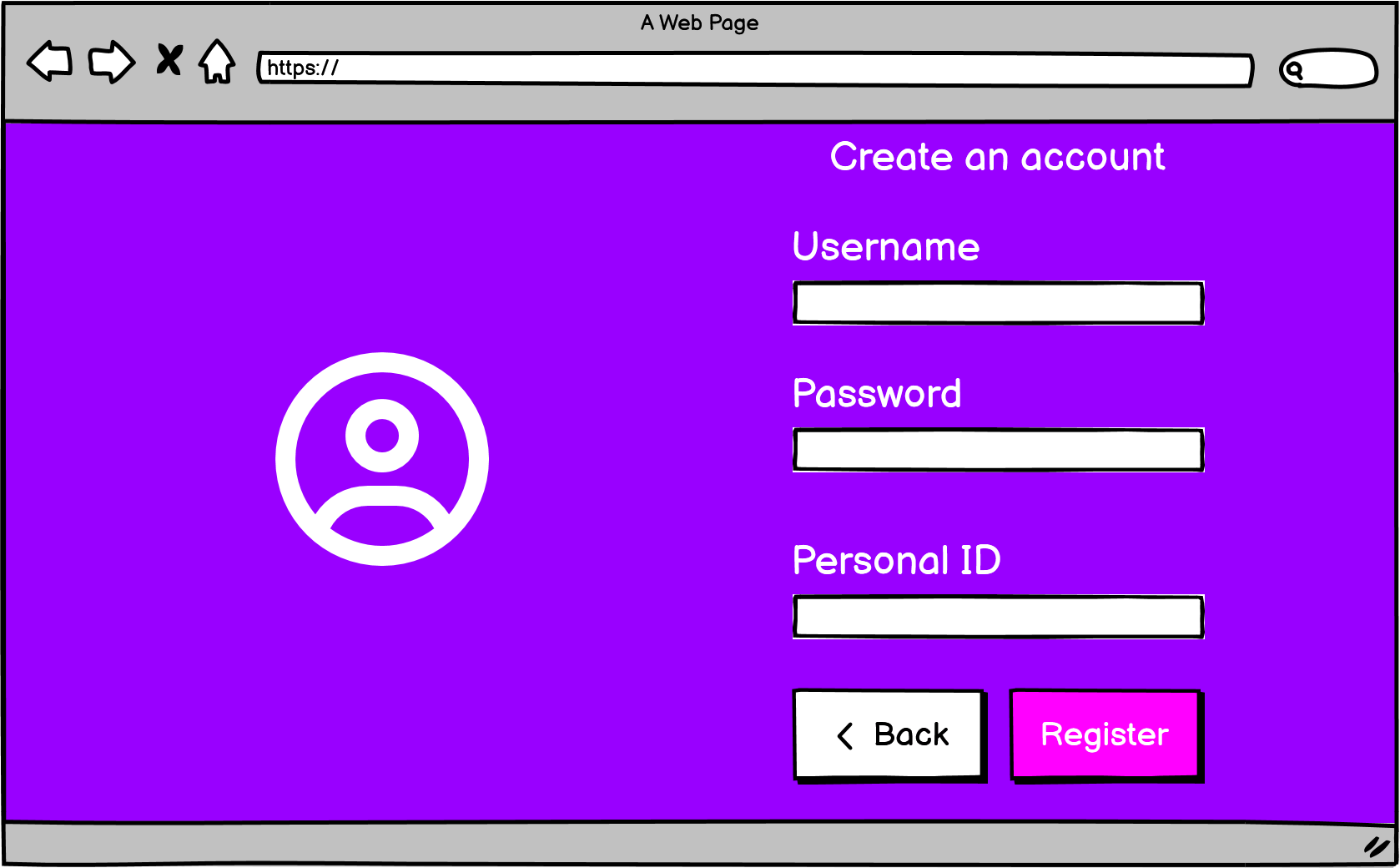


1. Design an either wireframe/mockup with balsamiq or prototype with figma for your use cases.

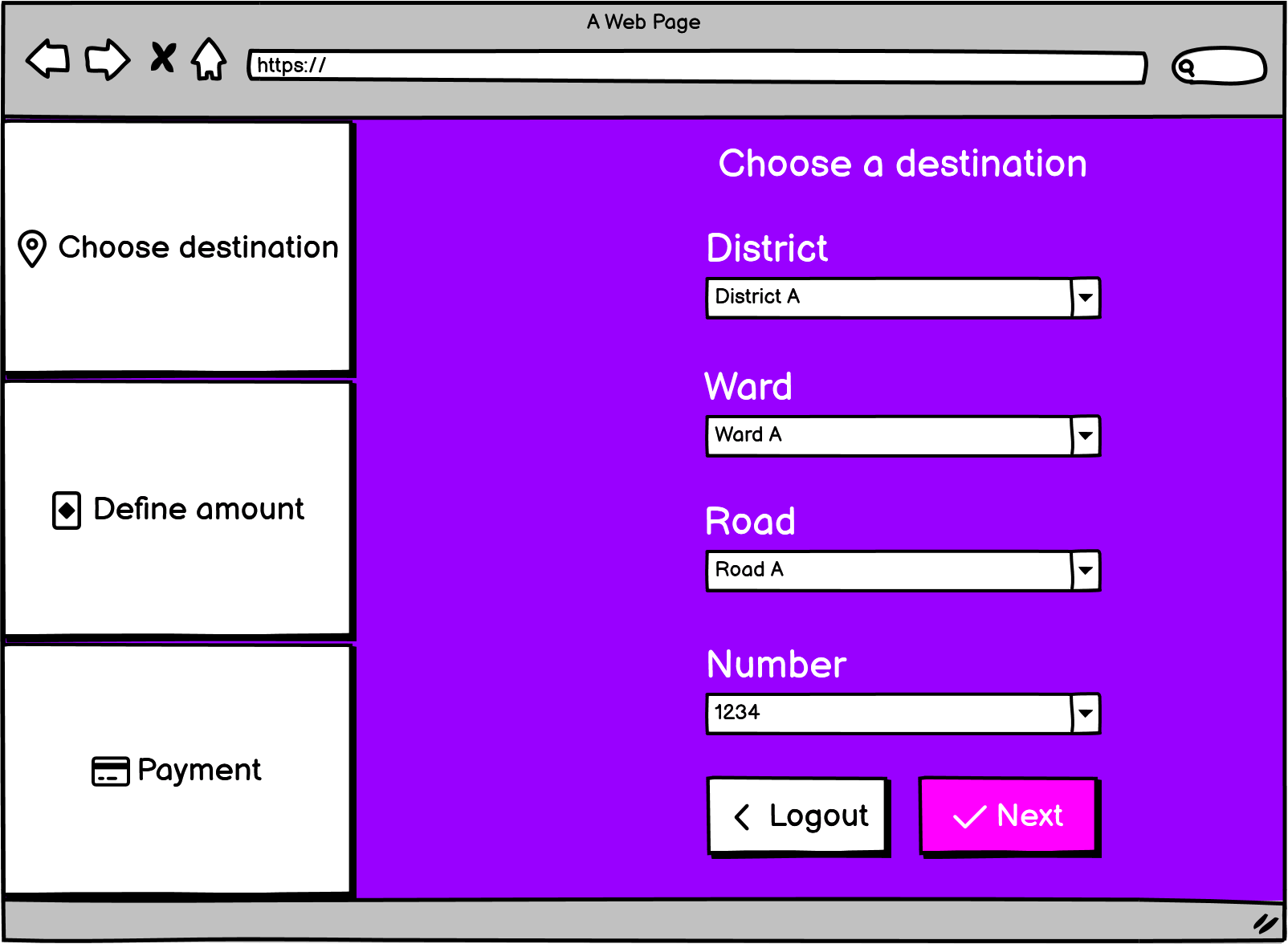
* Login screen



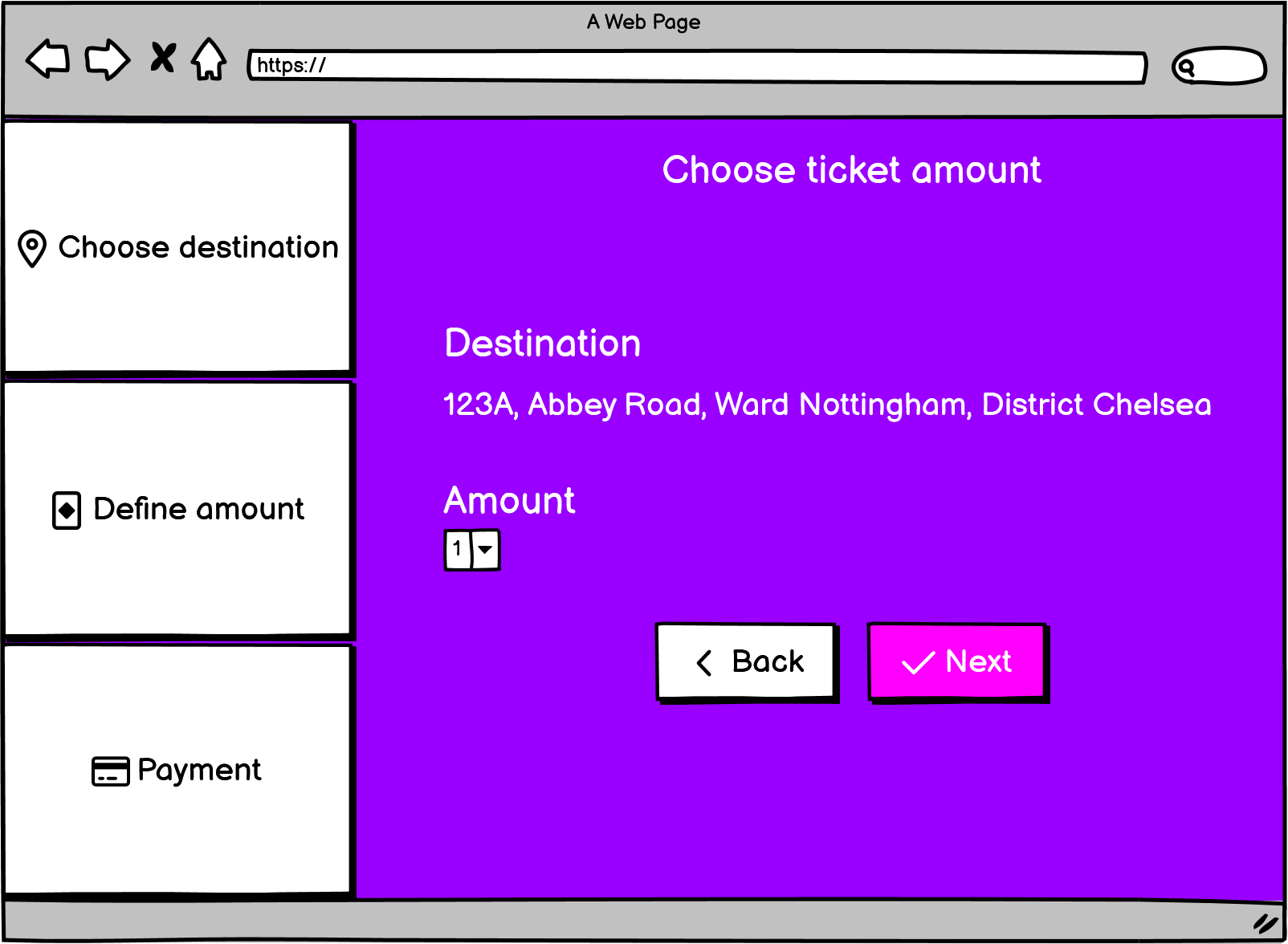
* Register screen



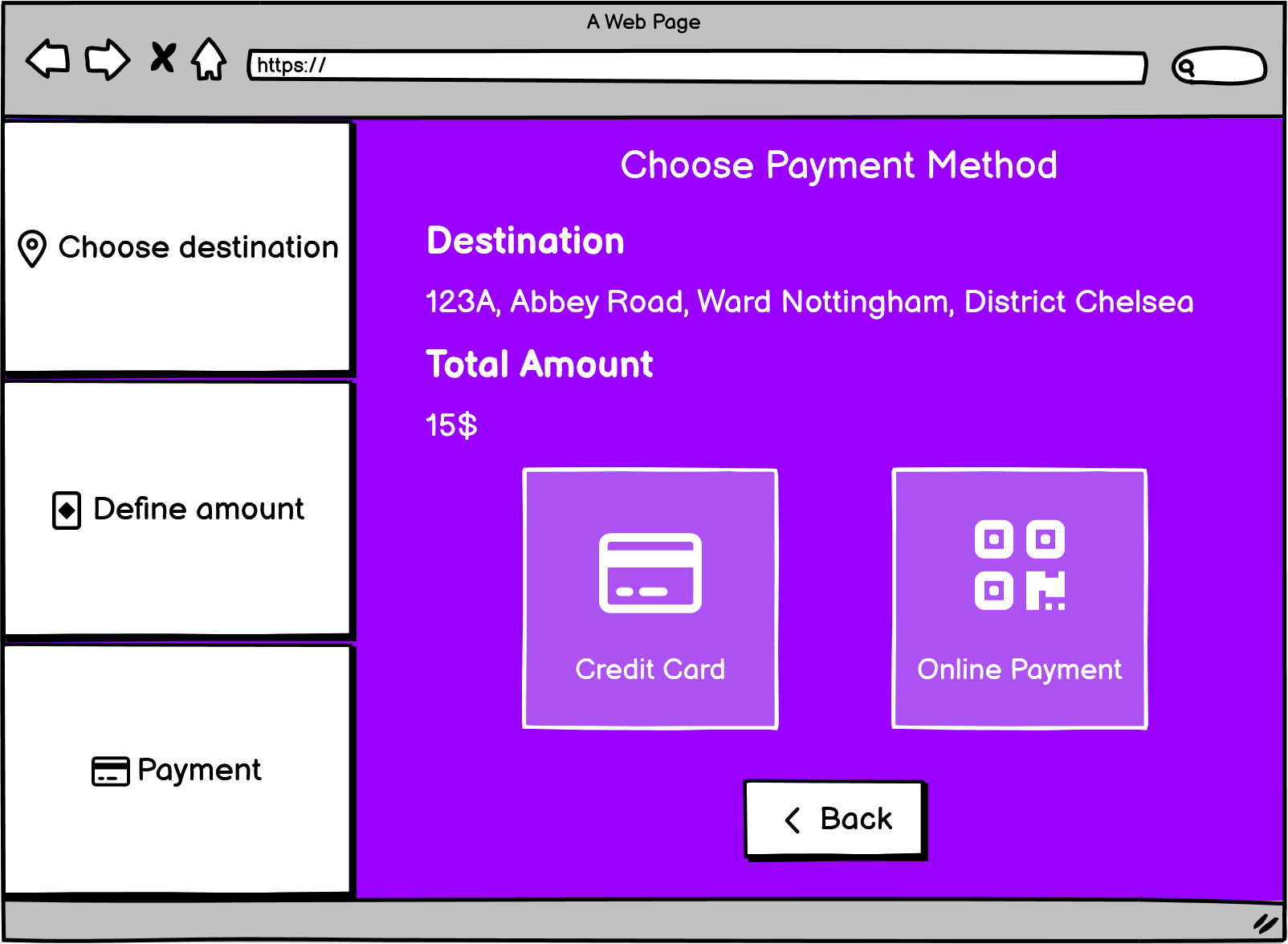
* Destination picker page



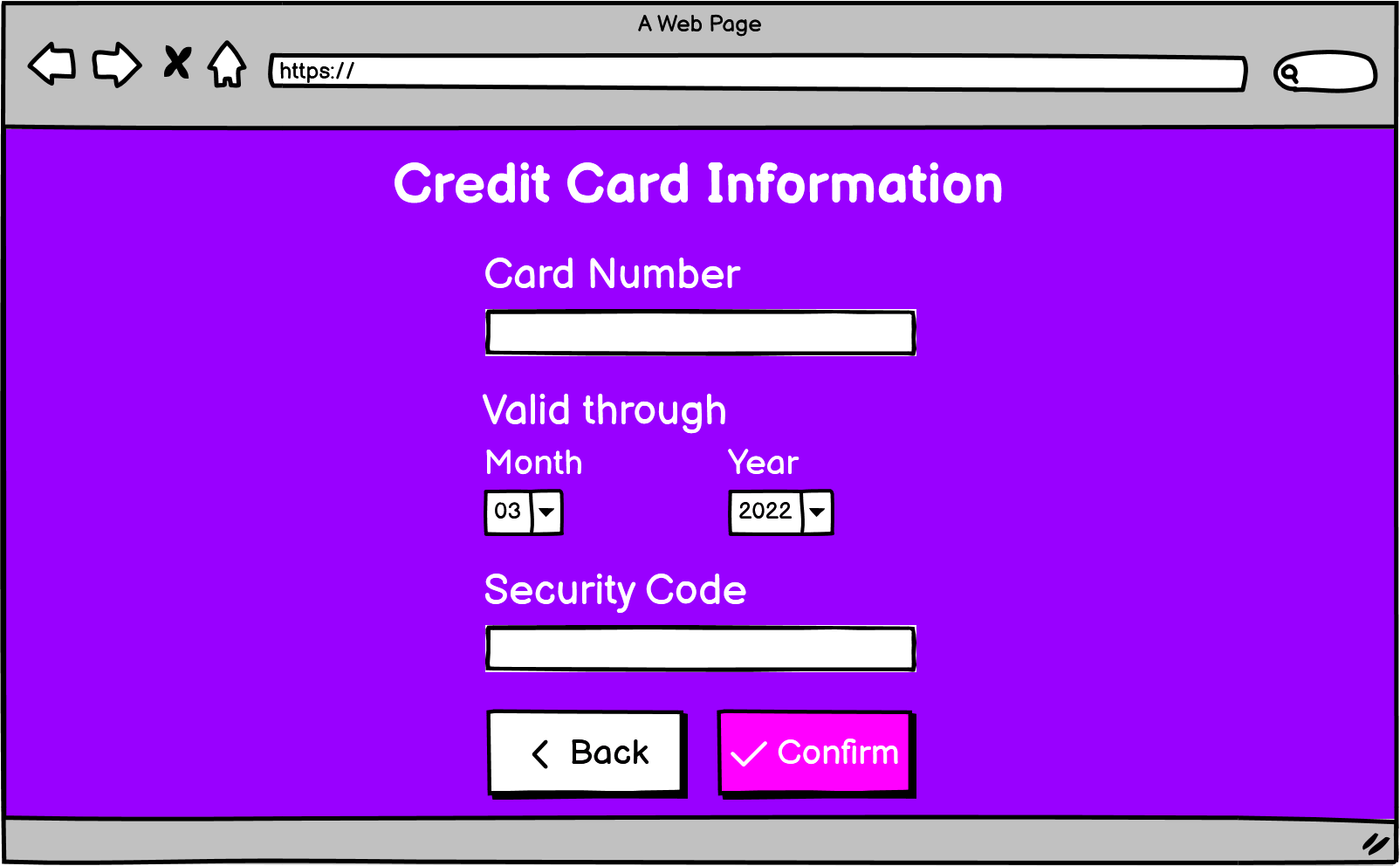
* Ticket amount picket page



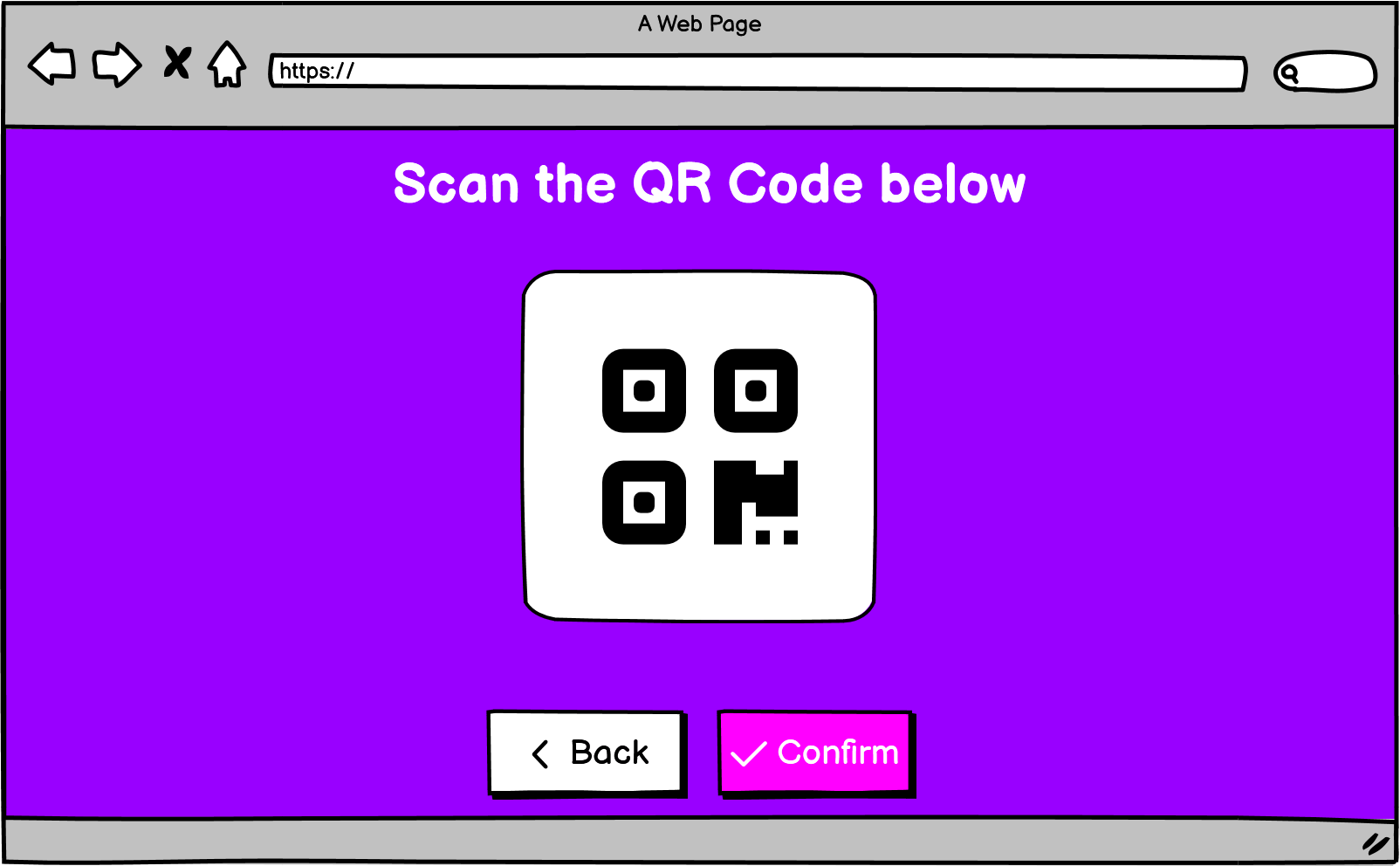
* Payment page



* Credit card payment page



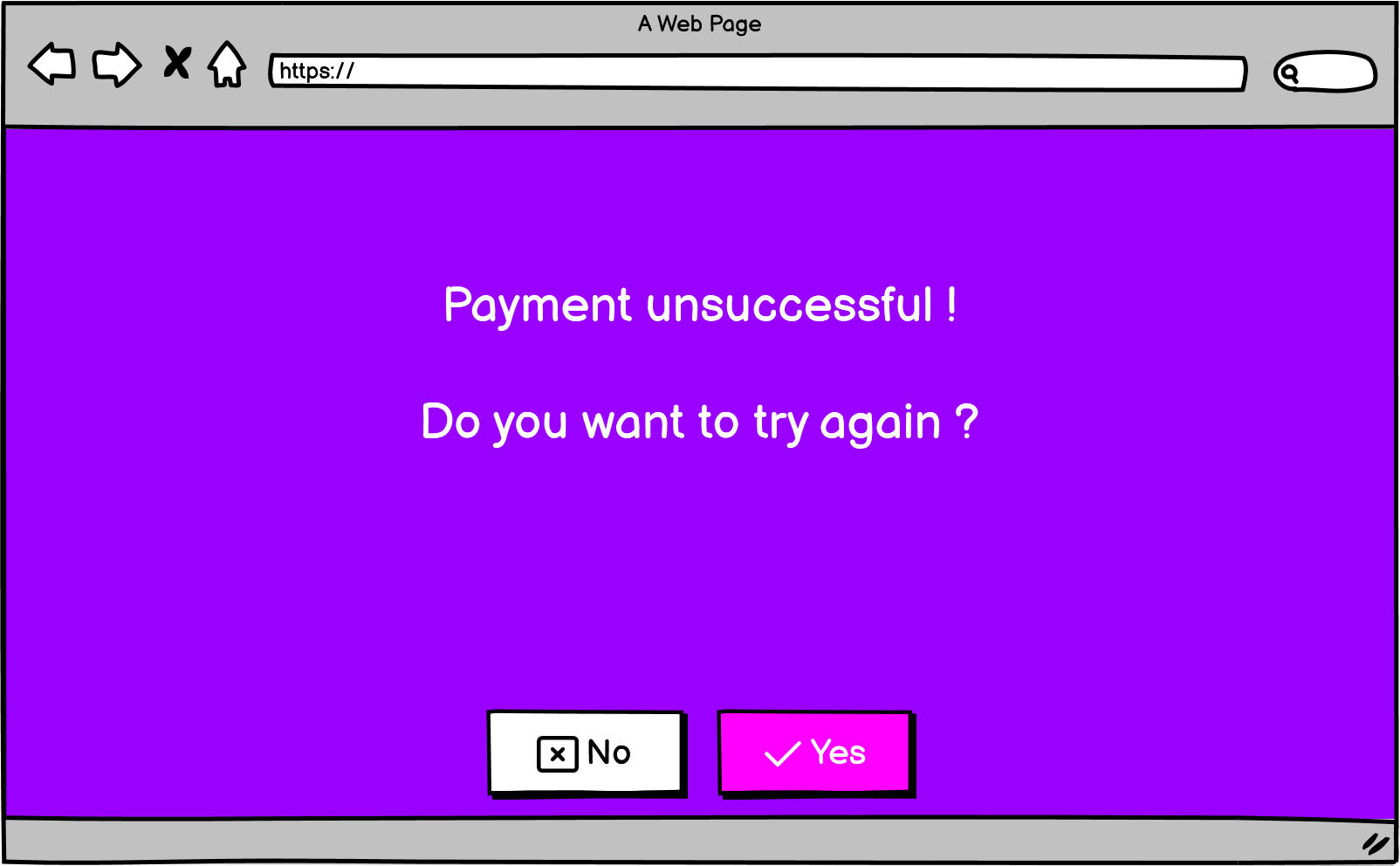
* QR Code payment page



* Successful payment page

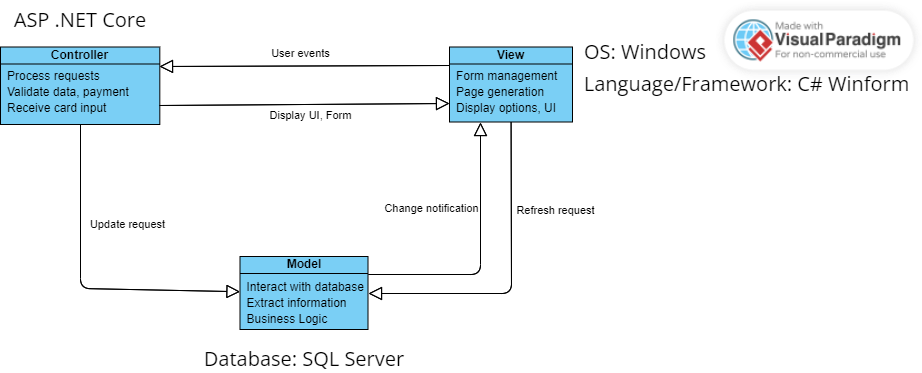


* Unsuccessful payment page

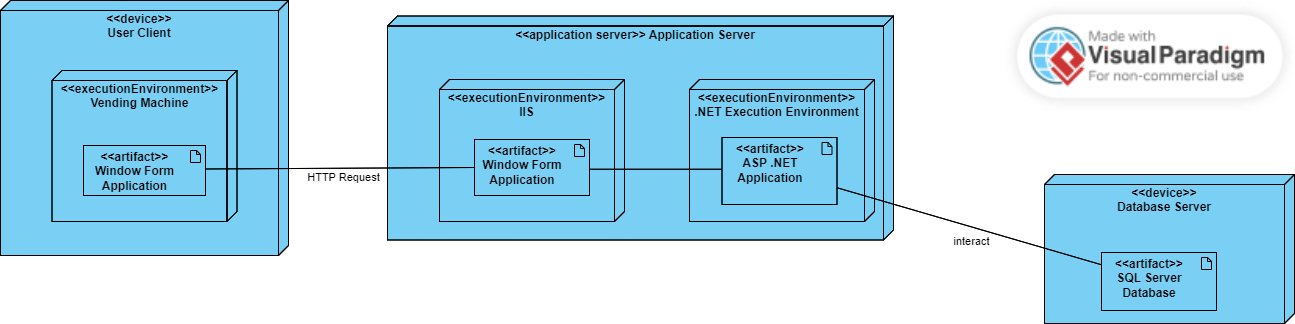


1. Develop Architecture design (System in-a-box or MVC model) and Deployment diagram for Ticket Vendor Machine.

* MVC Model

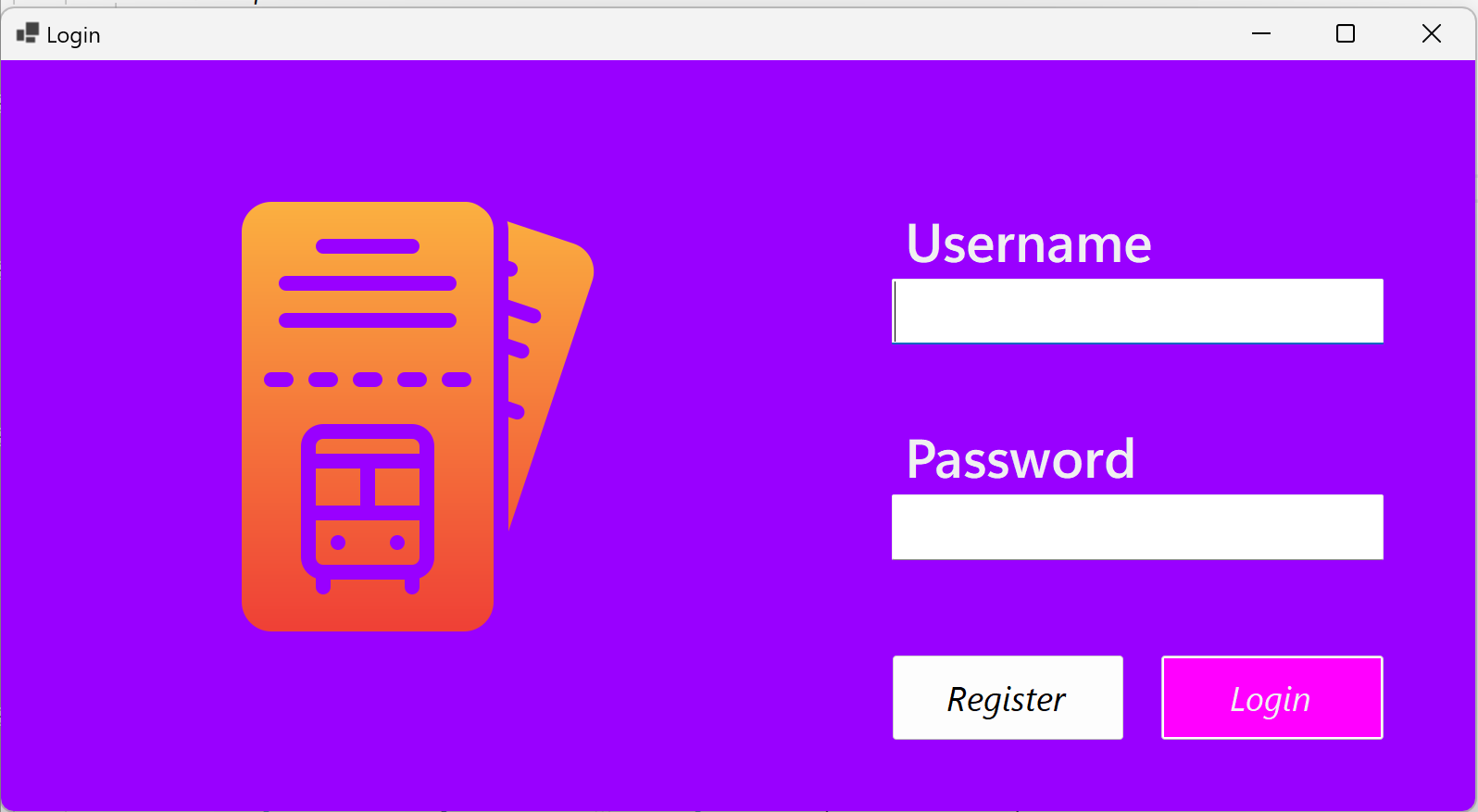


* Deployment diagram

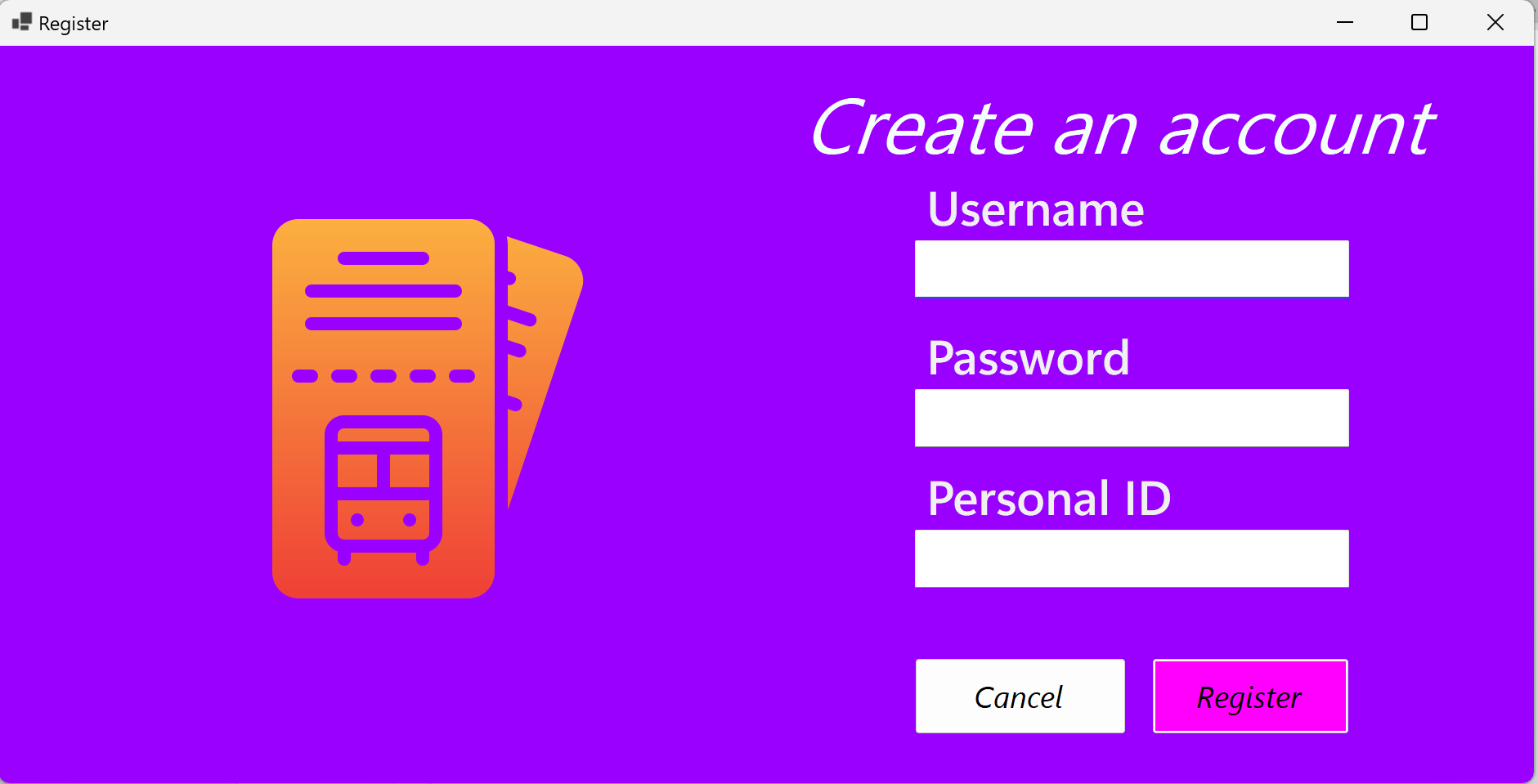


1. Demo any use case (form for inputs, report for output) with Visual Programming C# and MSSQL.

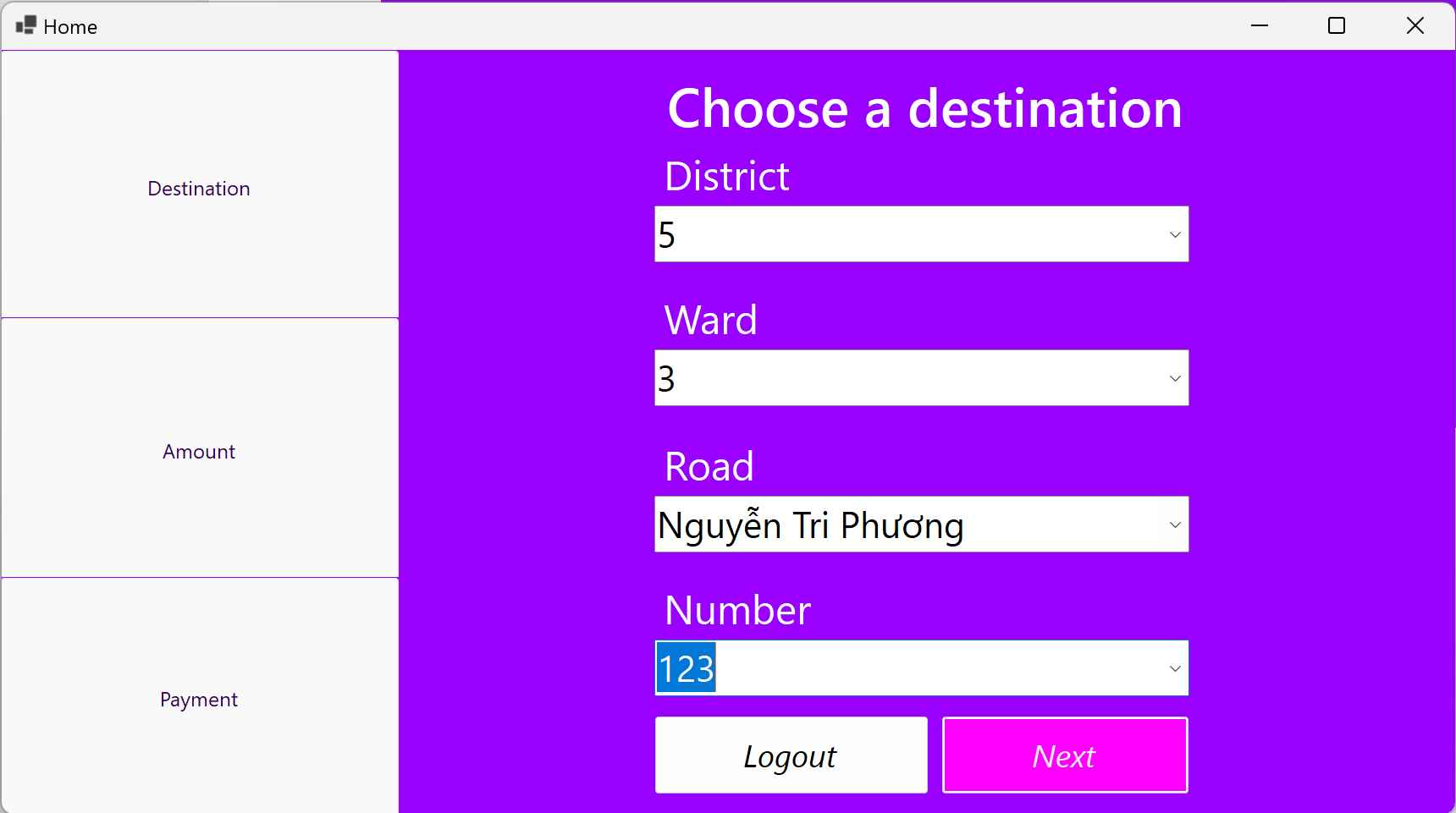
* Login page



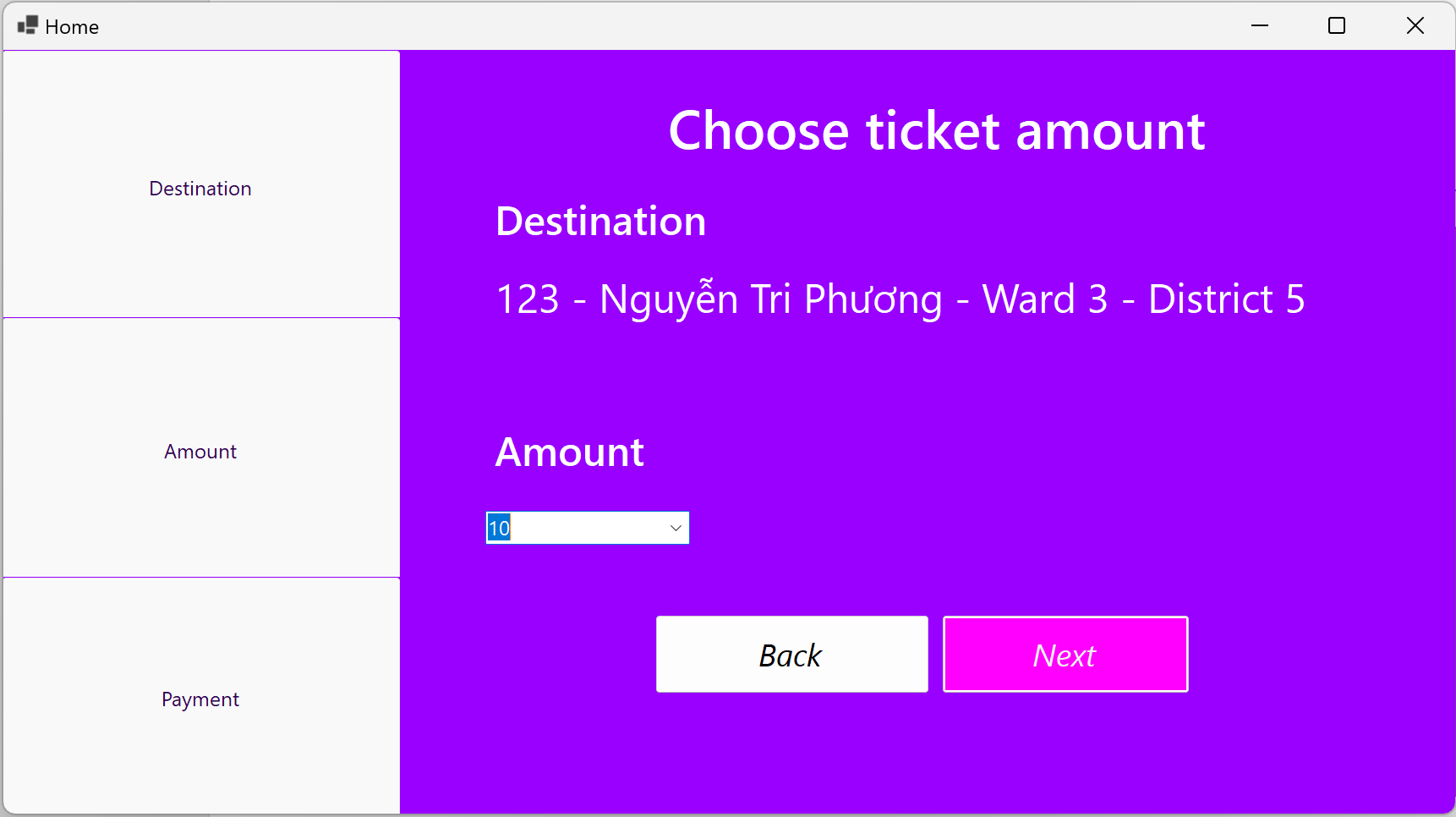
* Register page



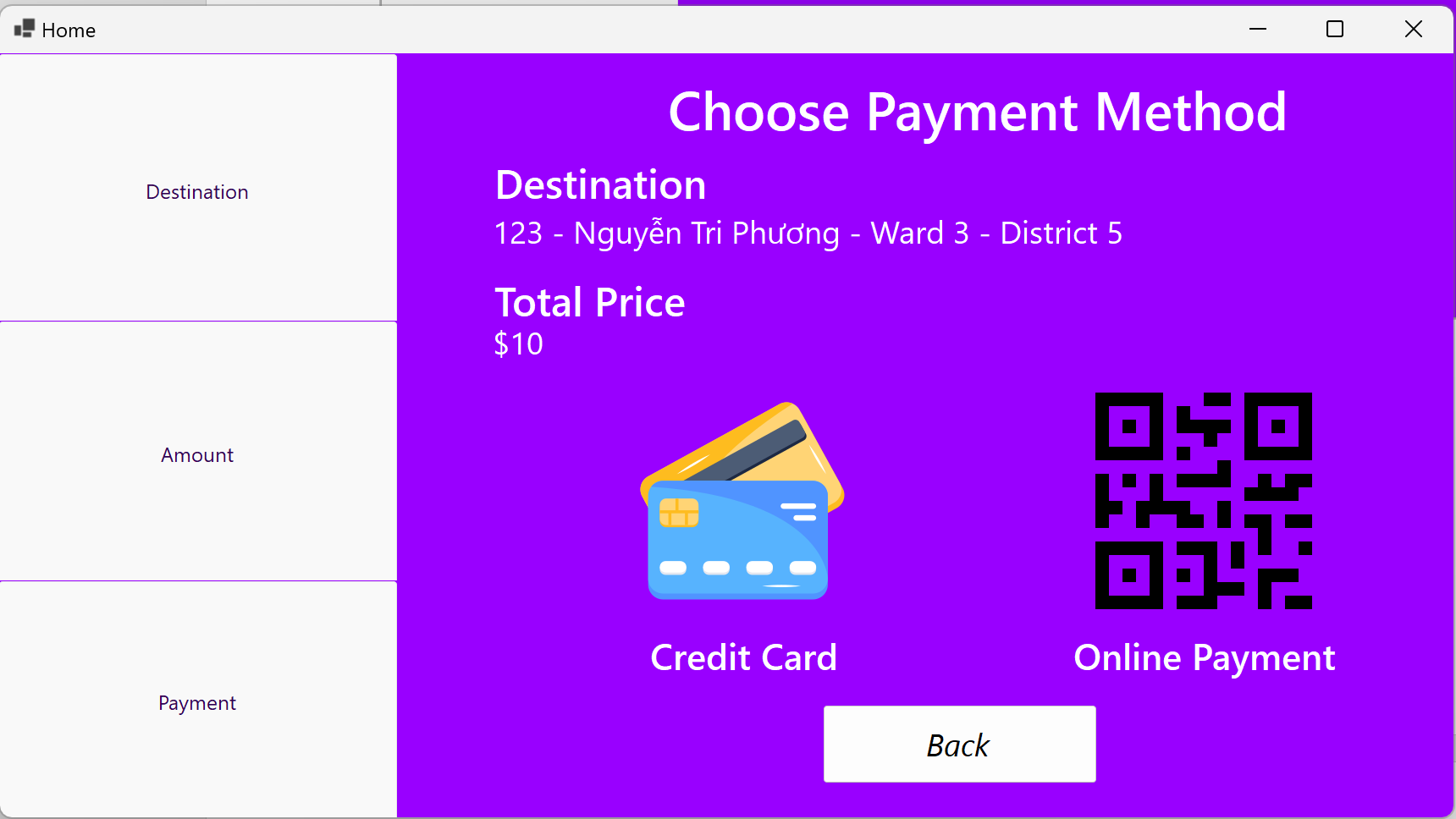
* Destination selection page



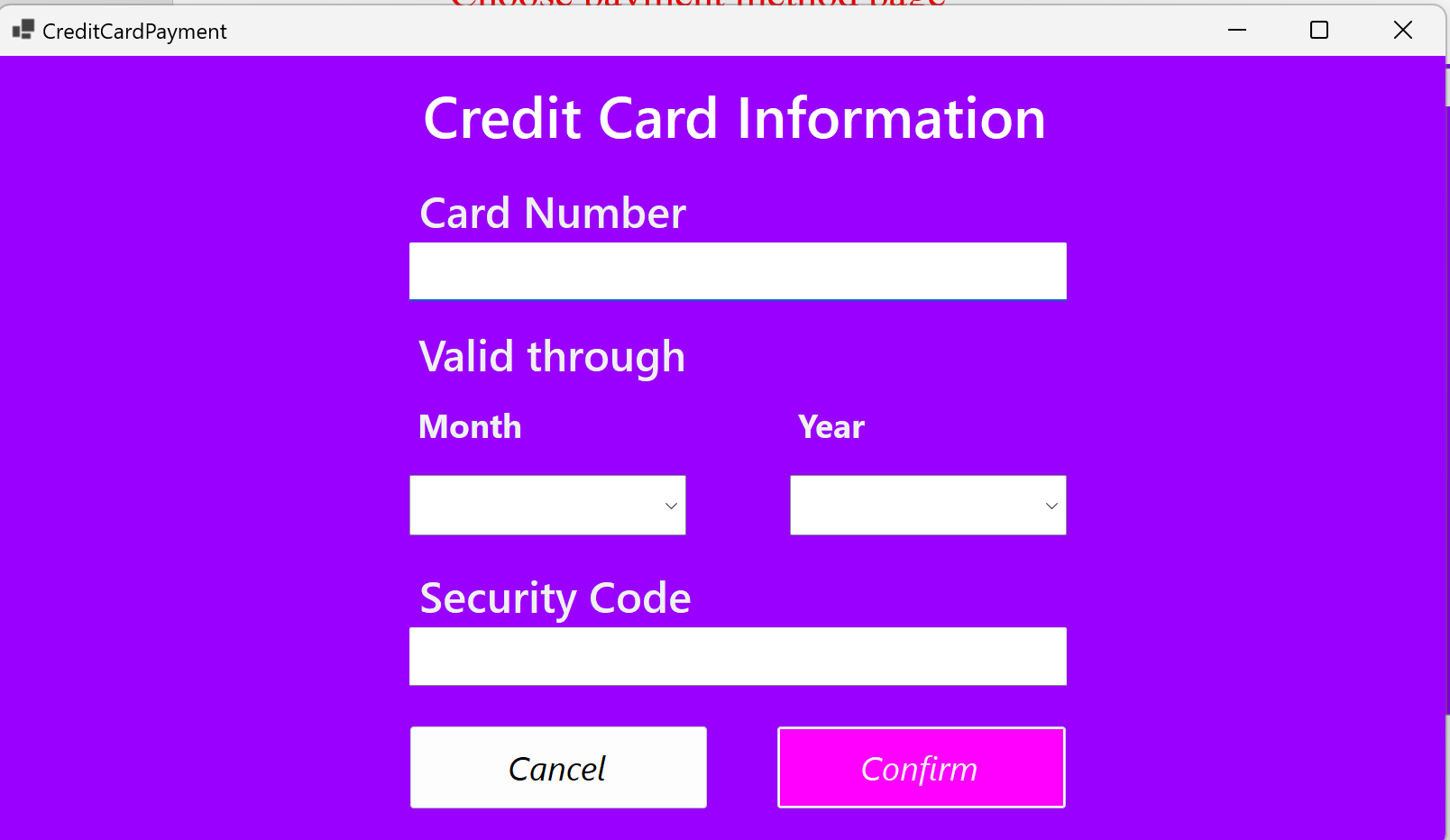
* Ticket amount picker page



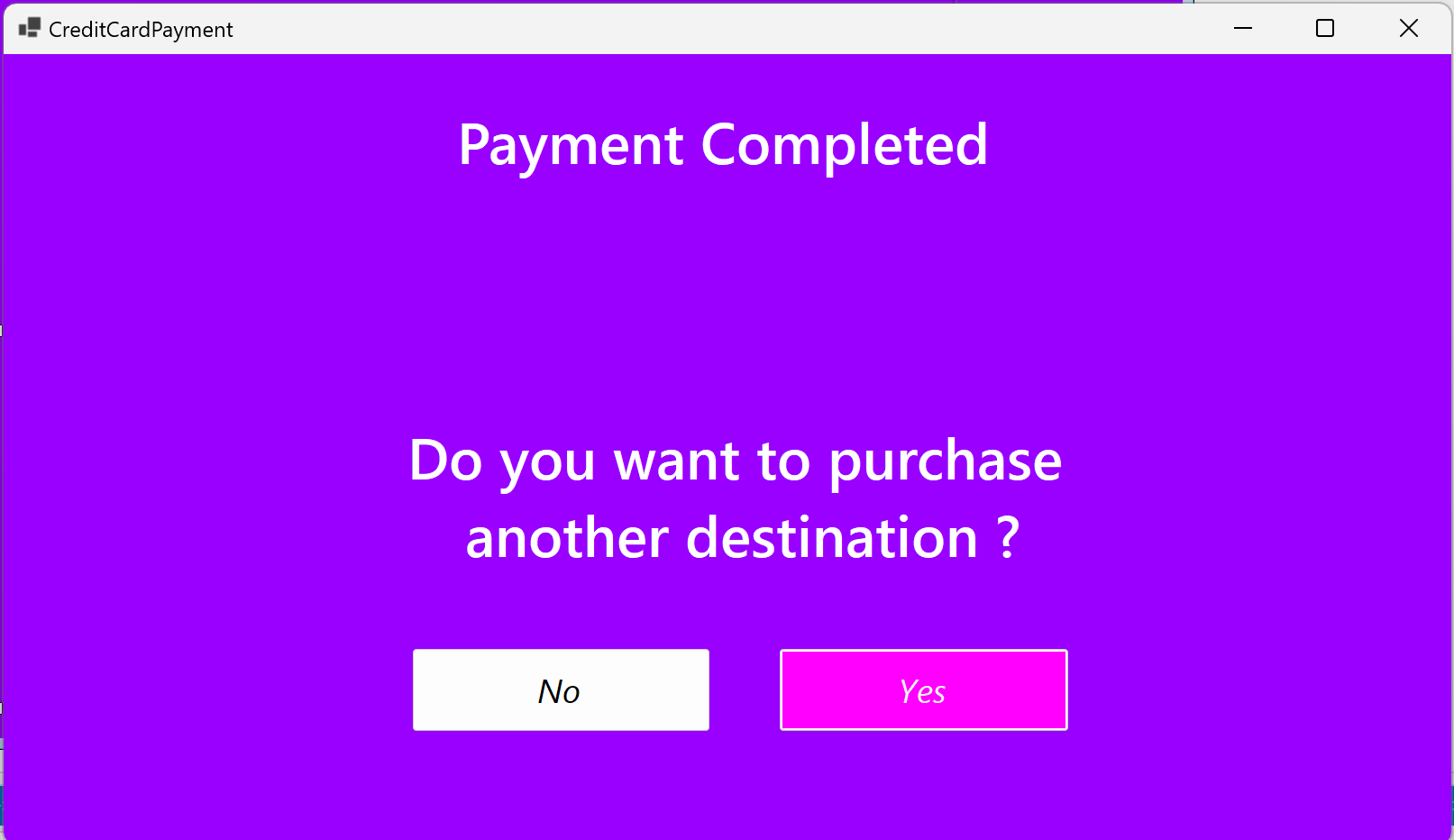
* Choose payment method page



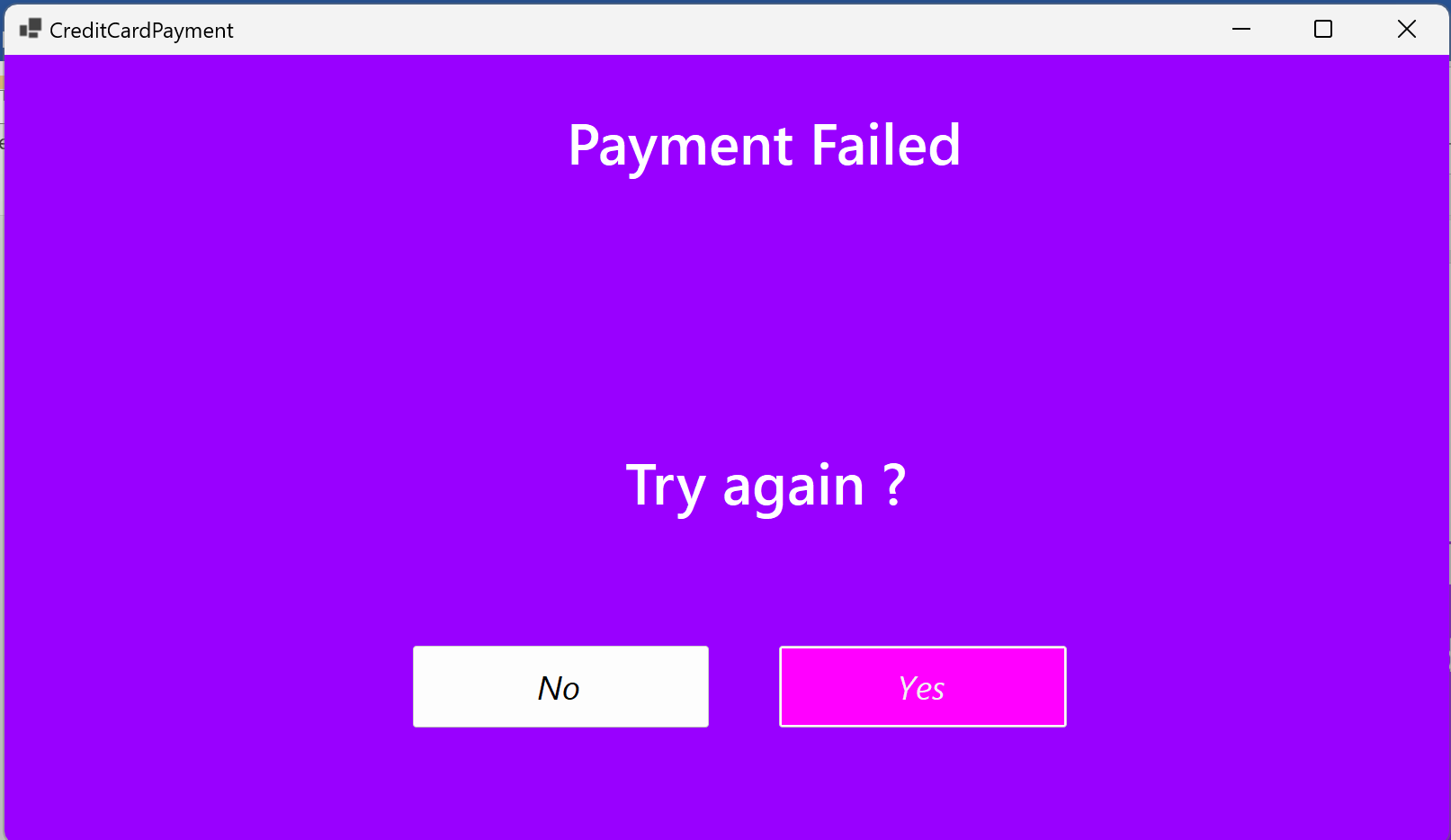
* Credit card inputs form



* Message if payment is successful



* Message if payment is unsuccessful

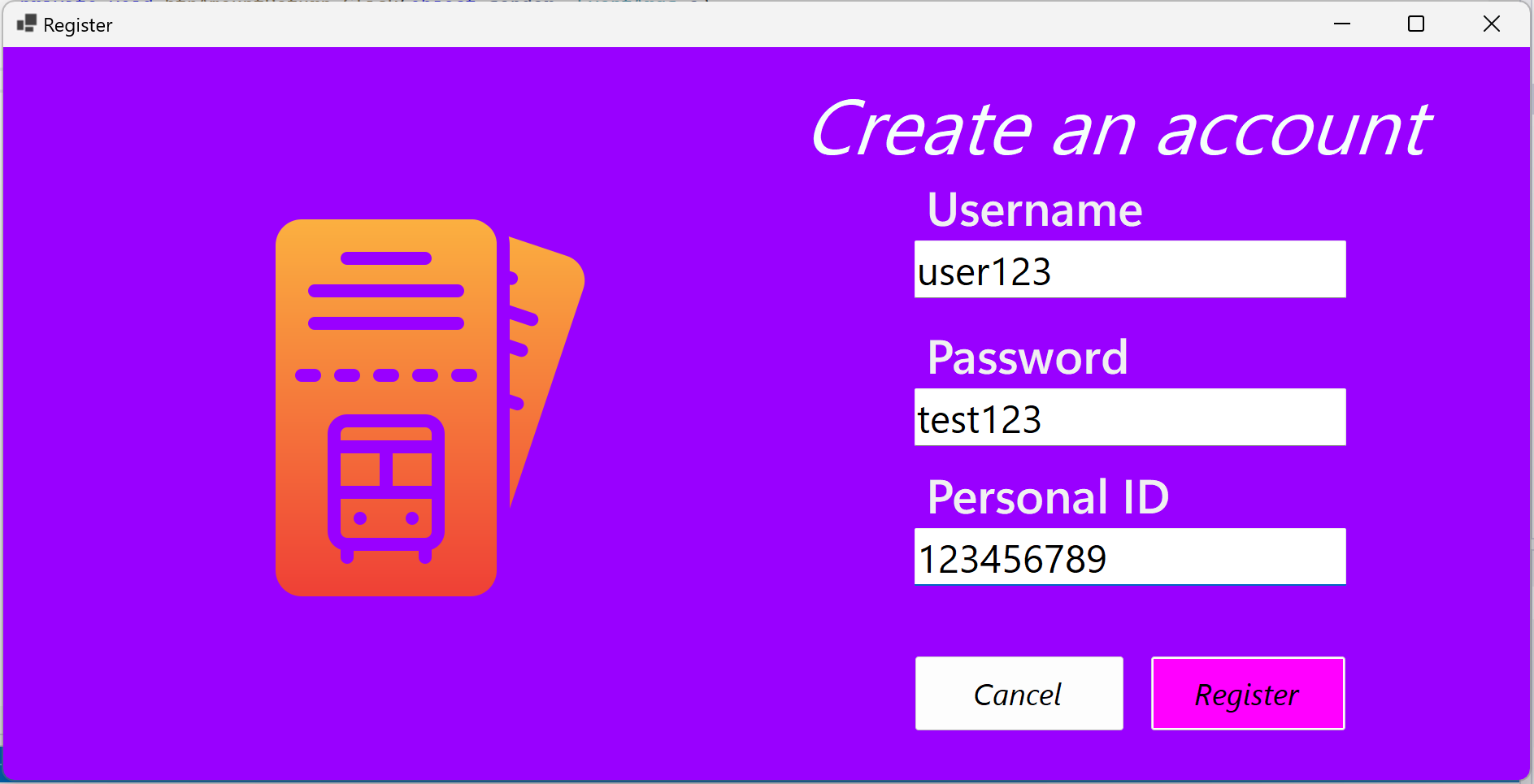


* Online payment QR Code Screen

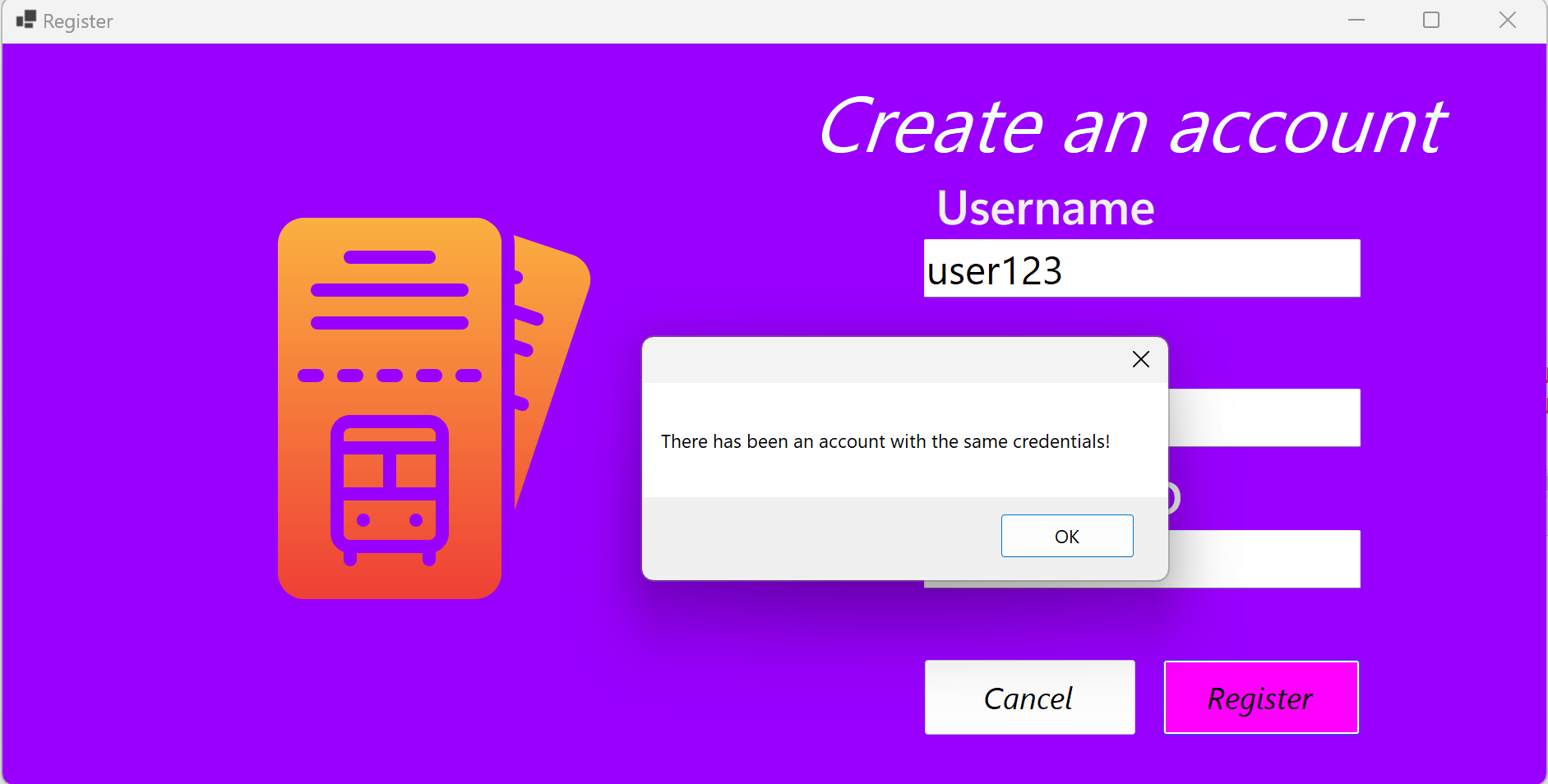


* The messages displaying the payment status are the same as the ones used for credit card payment

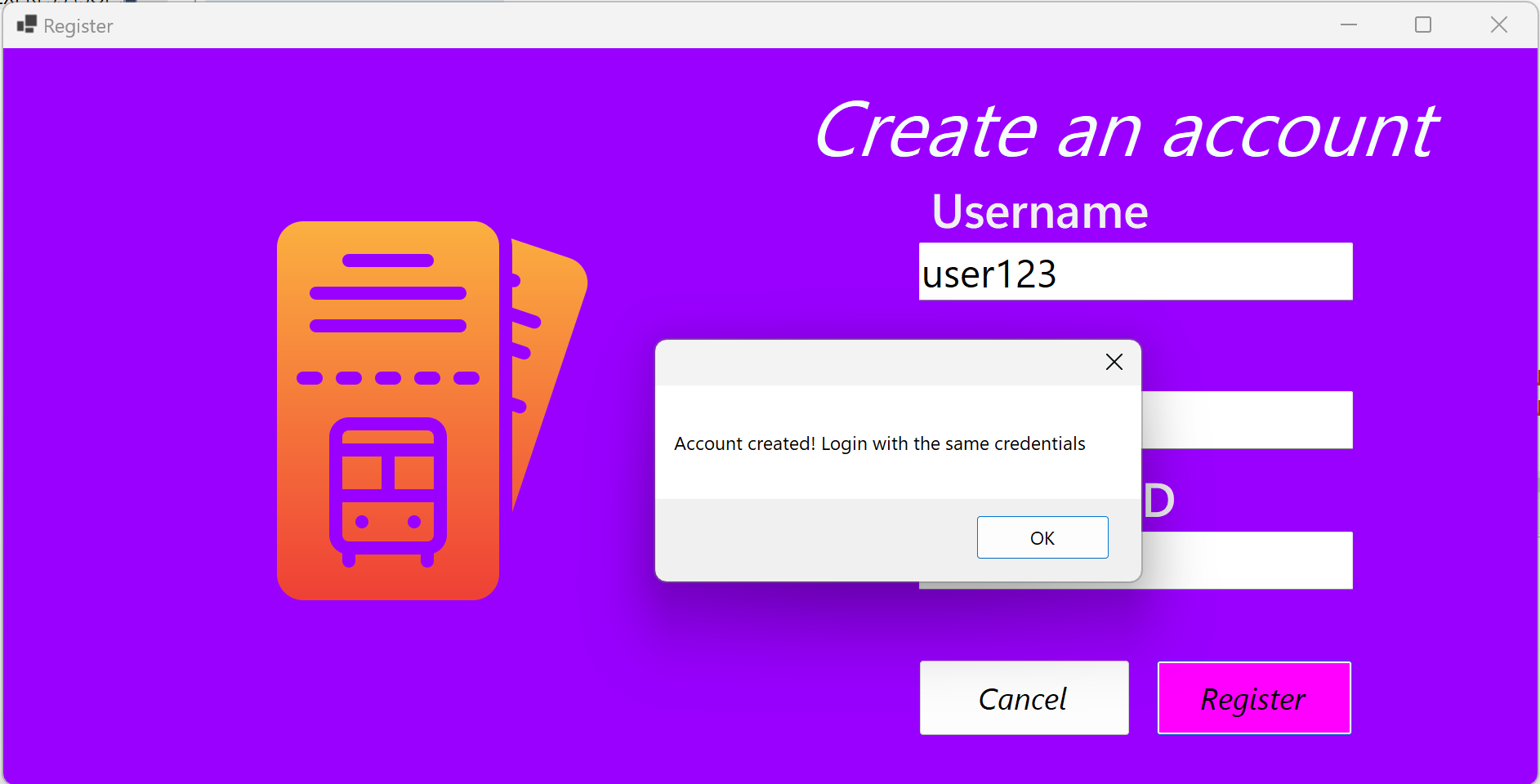
Here is an example of the account registration use case. First, the user has to enter their username, personal id and password.



The user then should press the Register button. If the username or the personal id have already been used, a failed registration message is displayed.



Alternatively, a successful registration message is displayed when the username nor the personal id was previously used.



If the user did not enter all or any of the specific fields. A request to fill them in is shown to the user.

