|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Actor** | **Use Case Description** | **Dependancy** |
| Delete Card | Client | A client deletes a card entry on record | EXTENDED by noDebitCardOnRecord |
| Add Card |  | A client adds a card entry on record |  |
| Withdraw cash from debit |  | A client withdraw cash from their default debit card |  |
| Borrow cash |  | A client wants to borrow cash from our service |  |
| View transaction |  | A client wants to view an incomplete transaction or a transaction in progress. |  |
| updatePersonalInformation |  | A client wants to change their personal information such as name, email, etc |  |
| ChangeDefaultCard |  | A client wants to change the default card that is charged automatically for withdrawal and payment of loans. |  |
| Pickupcash | N/A | A client wants to pickup cash from an agent | EXTENDED by updateMyCurrentLocation  INCLUDED by ShowNearestAgents |
| RequestDelivery | N/A | A client request for an agent to deliver cash | EXTENDED by updateMyCurrentLocation , createCashDeliveryOrder |
| cancelCashOrder | N/A | A client decides to cancel a cash order | N/A |
| cancelDeliveryOrder | N/A | A client wants to cancel a delivery order | N/A |
| showNearestAgent | N/A | A system shows the client the nearest cash agents | INCLUDED by pickCashAgent, updateMyCurrentLocation  computeNearestAgent |
| pickCashAgent | N/A | A client picks |  |

|  |  |
| --- | --- |
| **Case Name** | Delete Card |
| **Brief Description** | A client deletes a debit card on record |
| **Precondition** | Client is on profile page. |
| **Primary Actor** | Client |
| **Secondary Actor** | Database |
| **Dependency** | EXTENDED by noCardOnRecord |
| **Generalization** | None |

|  |
| --- |
| **Basic Flow** |
| |  |  | | --- | --- | | **STEP #** |  | | 1 | The system gets a list of stored cards on record from the database and display them on the screen | | 2 | The client selects the card that they want to remove from the database record | | 3 | The client clicks delete | | 4 | The system makes request to the stripe api to remove the card from record | | 4 | The system deletes the stored card from the database record | | 6 | The system shows a confirmation page to the client that their card has been successfully deleted from the record | | **Post Condition** | Client is on Profile Page | |

|  |
| --- |
| **Specific Alternative Flows** |
| |  |  | | --- | --- | | **RFS** | **Steps** | | RFS Basic Flow 1 | |  |  | | --- | --- | | **STEPS #1** |  | | 1 | The system gets 0 record of card from the database. | | 2 | The system displays a response that there is no card that has been added. | | **Post Condition** | Client is on profile page | | |

|  |  |
| --- | --- |
| **Use Case Name** | Add debit card |
| **Brief Description** | Client wants to add a card on record |
| **Precondition** | Client is on profile page. Client clicks add button. |
| **Primary Actor** | Client |
| **Secondary Actors** | Database, Stripe API |
| **Dependency** | None |
| **Generalization** | None |

|  |
| --- |
| **Basic Flow** |
| |  |  | | --- | --- | | **STEP #** |  | | 1 | System displays Stripe Payment Form using Stripe.js | | 2 | Client enters card information | | 3 | Client clicks submit button | | 4 | The Stripe.js sends request to the Stripe Servers | | 5 | A card token key is returned from the Stripe Servers and passed to the system on the client side | | 6 | The system then sends the token key to the server to be processed | | 7 | The server finds Stripe customer id from the database record | | 8 | The server then passes the Stripe customer id with the card token to the Stripe API to be saved | | 9 | The Stripe API returns a card object | | 10 | The server then extrapolates the card ID, last4digits, and brand. The server then stores it into the database record | | **Post Condition** | The Profile page is displayed. | |

|  |
| --- |
| **Specific Alternative Flows** |
| |  |  | | --- | --- | | **RFS** | **STEPS** | | RFS Basic Flow 7 | |  |  | | --- | --- | | **STEPS #** |  | | 1 | The Server does not find stripe customer id from the database | | 2 | The Server gets customer’s email from the database record | | 3 | The Server then request the Stripe Api to create a Stripe Customer ID | | 4 | The Stripe API returns a customer object | |  | The Server extract Stripe Customer ID from the customer object | | 5 | The Server stores the Stripe Customer ID into the database | | Post Conditon | Return to Basic Flow step 8 | | |