|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / doAction  Intended action: Performs a specific action based on the selection box and catches invalid errors. Invalid errors prevent execution. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to show an error when trying to close a nonexistent account. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s, attempt to close any account when the database is empty or when the said profile was not used. | ERROR (FirstName LastName DOB accountType) is not in the database. | ERROR (FirstName LastName DOB accountType) is not in the database. |
| 2 | The GUI must be able to show an error when closing an already closed account. | When a given name in “FirstName LastName” format and a date of birth whose date is before today’s is of a specific account type (like savings) and is already closed, attempt to close this account using the same name, date of birth, and account type. | ERROR Account is closed already. | ERROR Account is closed already. |
| 3 | The GUI is able to close an existing account. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s has an account (like savings) in the database, attempt to close this account by using the close option and inputting the same name, dob, and account type in the fields. | Account closed. | Account closed. |
| 4 | The GUI must be able to show an error when depositing into a nonexistent account. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s, attempt to deposit into any account when the database is empty or when the said profile was not used. | ERROR (FirstName LastName DOB accountType) is not in the database. | ERROR (FirstName LastName DOB accountType) is not in the database. |
| 5 | The GUI must be able to show an error when trying to deposit into a closed account. | When a given name in “FirstName LastName” format and a date of birth whose date is before today’s is of a specific account type (like savings) and is already closed, attempt to deposit into this account using the same name, date of birth, and account type. | ERROR Can’t deposit into closed account. | ERROR Can’t deposit into closed account. |
| 6 | The GUI is able to deposit into an existing account. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s has an account (like savings) in the database, attempt to deposit into this account by using the deposit option and inputting the same name, dob, and account type in the fields. | Deposit – balance updated. | Deposit – balance updated. |
| 7 | The GUI must be able to show an error when withdrawing from a nonexistent account. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s, attempt to withdraw from any account when the database is empty or when the said profile was not used. | ERROR (FirstName LastName DOB accountType) is not in the database. | ERROR (FirstName LastName DOB accountType) is not in the database. |
| 8 | The GUI must be able to show an error when withdrawing from a closed account. | When a given name in “FirstName LastName” format and a date of birth whose date is before today’s is of a specific account type (like savings) and is already closed, attempt to withdraw from this account using the same name, date of birth, and account type. | ERROR Can’t withdraw from closed account. | ERROR Can’t withdraw from closed account. |
| 9 | The GUI must be able to show an error when withdrawing from an account with insufficient funds. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s has an account (like savings) in the database, attempt to withdraw from this account by using the withdraw option and inputting the same name, dob, and account type in the fields. Then use a value greater than its current balance as the amount to withdraw from. | ERROR Withdraw – insufficient fund. | ERROR Withdraw – insufficient fund. |
| 10 | The GUI is able to withdraw from an existing account, where the amount withdrawn is less than the account’s current balance. | Given a name in “FirstName LastName” format and a date of birth whose date is before today’s has an account (like savings) in the database, attempt to withdraw from this account by using the withdraw option and inputting the same name, dob, and account type in the fields. Then use a value less than its current balance as the amount to withdraw from. | Withdraw – balance updated. | Withdraw – balance updated. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / checkingAccount  Intended action: Opens a checking account and handles duplicate errors or closed problems. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to find a duplicate account and show an error. | Open a checking account with a given name and date of birth (date before today), then attempt to open another checking account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 2 | The GUI must be able to find a college checking account under the same profile and show an error. | When a college checking account of a given name and date of birth (date before today) is already in the database, attempt to open a checking account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 3 | The GUI must be able to reopen a closed account. | When a checking account of a given name and date of birth (date before today) was closed in some previous action, open a checking account with the same name and date of birth. | Account reopened. | Account reopened. |
| 4 | The GUI must be able to open a new account. | Attempt to open a checking that consists of a name in the “FirstName LastName” format and a date of birth that is before today. | Account opened. | Account opened. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / collegeCheckingAccount  Intended action: Opens a college checking account and handles duplicate errors or closed problems. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to find a duplicate account and show an error. | Open a college checking account with a given name and date of birth (date before today), then attempt to open another college checking account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 2 | The GUI must be able to find a checking account under the same profile and show an error. | When a checking account of a given name and date of birth (date before today) is already in the database, attempt to open a college checking account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 3 | The GUI must be able to reopen a closed account. | When a college checking account of a given name and date of birth (date before today) was closed in some previous action, open a college checking account with the same name and date of birth. | Account reopened. | Account reopened. |
| 4 | The GUI must be able to open a new account. | Attempt to open a college checking that consists of a name in the “FirstName LastName” format and a date of birth that is before today. | Account opened. | Account opened. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / savingsAccount  Intended action: Opens a saving account and handles duplicate errors or closed problems. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to find a duplicate account and show an error. | Open a savings account with a given name and date of birth (date before today), then attempt to open another savings account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 2 | The GUI must be able to reopen a closed account. | When a savings account of a given name and date of birth (date before today) was closed in some previous action, open a savings account with the same name and date of birth. | Account reopened. | Account reopened. |
| 3 | The GUI must be able to open a new account. | Attempt to open a savings that consists of a name in the “FirstName LastName” format and a date of birth that is before today. | Account opened. | Account opened. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / moneyMarketAccount  Intended action: Opens a Money Market account and handles duplicate errors or closed problems. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to find a duplicate account and show an error. | Open a money market account with a given name and date of birth (date before today), then attempt to open another money market account of the same name and date of birth. | ERROR (FirstName LastName DOB) same account(type) is in the database. | ERROR (FirstName LastName DOB) same account(type) is in the database. |
| 2 | The GUI must be able to reopen a closed account. | When a money market account of a given name and date of birth (date before today) was closed in some previous action, open a money market account with the same name and date of birth. | Account reopened. | Account reopened. |
| 3 | The GUI must be able to open a new account. | Attempt to open a money market that consists of a name in the “FirstName LastName” format and a date of birth that is before today. | Account opened. | Account opened. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / checkBalance  Intended action: Checks if the money being put into an account is non-negative and is a number. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to reject and show an error with a 0 or negative initial deposit (opening an account). | When opening a new account in the database, attempt to make the initial deposit any value equal or less than 0. | Initial deposit cannot be 0 or negative. | Initial deposit cannot be 0 or negative. |
| 2 | The GUI must be able to reject and show and error with a 0 or negative deposit amount. | When depositing into an existing account in the database, attempt to deposit any value equal or less than 0. | Deposit – amount cannot be 0 or negative. | Deposit – amount cannot be 0 or negative. |
| 3 | The GUI must be able to reject and show an error when depositing less than the minimum required for a money market account. | When opening a new money market account, attempt to make the initial deposit less than the necessary balance ($2500 for this case) | Minimum of $(Balance needed to open a Money Market account) to open a MoneyMarket Account | Minimum of $(Balance needed to open a Money Market account) to open a MoneyMarket Account |
| 4 | The GUI must be able to reject and show an error when an initial balance for a new account has a non-numerical value. | When opening a new account in the database, attempt to make the initial deposit any value that isn’t a number, like a letter. | ERROR: Initial deposit entry is not a valid amount. | ERROR: Initial deposit entry is not a valid amount. |
| 5 | The GUI must be able to reject and show an error when depositing a non-numerical value into an account. | When depositing into an existing account in the database, attempt to deposit any value that isn’t a number, like a letter. | ERROR: Deposit entry is not a valid amount. | ERROR: Deposit entry is not a valid amount. |
| 6 | The GUI must be able to properly deposit into an existing account given valid inputs. | When opening a new account in the database, attempt to make the initial deposit any value greater than 0. | Deposit – balance updated. | Deposit – balance updated. |
| 7 | The GUI must be able to properly open up an account given valid inputs. | When depositing into an existing account in the database, attempt to deposit any value greater than 0. | Account opened (if opening for the first time).  Account reopened (if reopening the account) | Account opened (if opening for the first time).  Account reopened (if reopening the account) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / checkCampusCode  Intended action: Checks if the campus code for a college checking account is non-negative, a number, and between 0 and 2. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to not show the campus code line if the account is not a college checking account. | When doing opening an account, use a name (FirstName LastName) and date of birth which has a date before today’s and select anything but College Checking. Verify the line isn’t there, then press back and select College Checking. Verify the line is there. | The Campus Code line only appears when selecting College Checking as the account type. | The Campus Code line only appears when selecting College Checking as the account type. |
| 2 | The GUI must be able to reject code values not between 0 and 2 and show an error regarding it. | On the Campus Code line, enter a value either greater than 2 or less than 0. | Invalid Campus Code. | Invalid Campus Code. |
| 3 | The GUI must be able to reject a non-numeric code value and show an error regarding it. | On the Campus Code line, enter a non-numeric value. | Invalid Campus Code. | Invalid Campus Code. |
| 4 | The GUI must be able to have no problems when opening a college checking account with a valid campus code. | On the Campus Code line, enter a numeric value that is one of three numbers: 0, 1, 2. | Account opened (if opening for the first time).  Account reopened (if reopening the account) | Account opened (if opening for the first time).  Account reopened (if reopening the account) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / checkLoyal  Intended action: Checks if the loyal code for a savings account is non-negative, a number, and either 0 or 1. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to not show the loyalty code line if the account is not a savings account. | When opening an account, use a name (FirstName LastName) and date of birth which has a date before today’s and select anything but Savings. Verify the line isn’t there, then press back and select Savings. Verify the line is there. | The Loyalty line shows up only when selecting the account type as savings. | The Loyalty line shows up only when selecting the account type as savings. |
| 2 | The GUI must be able to reject a loyalty code that is not 0 or 1 and show an error for it. | On the Loyalty Code line, enter a value either greater than 1 or less than 0. | Invalid loyalty code. | Invalid loyalty code. |
| 3 | The GUI must be able to reject a non-numeric loyalty code and show an error for it. | On the Loyalty Code line, enter a non-numeric value. | Invalid loyalty code. | Invalid loyalty code. |
| 4 | The GUI must be able to have no problems in opening a savings account with a valid loyalty code. | On the Loyalty Code line, enter a numeric value that is either 0 or 1 | Account opened (if opening for the first time).  Account reopened (if reopening the account) | Account opened (if opening for the first time).  Account reopened (if reopening the account) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / getInfo  Intended action: Shows the necessary information based on the action desired. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI must be able to prompt the user to select an action if no action was selected. | Upon starting up the GUI, immediately press the Select button. | Please Select an Action | Please Select an Action |
| 2 | The GUI must be able to reveal the next set of panes (located in the middle) upon opening, closing, depositing, or withdrawing as their action. | Select Open Account, Close Account, Withdraw Money, or Deposit Money as the choice, then press the Select button. | The next set of panes will appear, showing a field for a name input, account type selection, and a date of birth input. | The next set of panes will appear, showing a field for a name input, account type selection, and a date of birth input. |
| 3 | The GUI must be able to print out the account database, whether it is empty or not and regardless of the print order selected. | Select Display Database, Display Database by account type, or Display accounts with fees and monthly interest, then press the Select button. | The printArea field (massive text box at the bottom) fills up with the account details if the database is filled, Account Database is empty! otherwise. | The printArea field (massive text box at the bottom) fills up with the account details if the database is filled, Account Database is empty! otherwise. |
| 4 | The GUI must be able to update the balance of the accounts in the database and say if it is empty or not. | Select Update Balances, then press the Select button. | Account Database is empty! will appear in the printArea (massive text box at the bottom) if the account database is empty, the account details with updated balances will appear otherwise. | Account Database is empty! will appear in the printArea (massive text box at the bottom) if the account database is empty, the account details with updated balances will appear otherwise. |
| 5 | The GUI must be able to stop upon selecting quit. | Select Quit, then press the Select button. | The GUI closes. | The GUI closes. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GuiController / populateProfile  Intended action: Creates a Profile based on the values given in the GUI. States all errors regarding the Profile, including invalid DOB or empty name. | | | | |
| Test Case # | Requirements | Test Description and Input Data | Expected Result / Output | Actual Result / Output |
| 1 | The GUI shall not proceed to the next step without any information. | When getting into the middle pane with nothing typed or selected, press the continue button without selecting anything. | Invalid name. Must be in format “FirstName LastName”. | Invalid name. Must be in format “FirstName LastName”. |
| 2 | The GUI requires the holder to have their name in “Firstname Lastname” format. | Type in only a first name in the field for the name, then press the continue button. | Invalid name. Must be in format “FirstName LastName”. | Invalid name. Must be in format “FirstName LastName”. |
| 3 | The GUI requires the holder to have their name in “Firstname Lastname” format. | Leave a space for the first name and a space to separate them. Type the last name then press the continue button. | Invalid name. Must be in format “FirstName LastName”. | Invalid name. Must be in format “FirstName LastName”. |
| 4 | The GUI requires an account to be selected to do an action. | When getting into the middle pane, have a valid name of the “FirstName LastName format” then press continue. | Must select account type. | Must select account type. |
| 5 | The GUI will not accept a date of birth that is today. | Select one of the account type options (savings, checking, college checking, or money market) and name inputted in “FirstName LastName” format. Set the date of birth to today’s date, then press continue. | Date of birth invalid. | Date of birth invalid. |
| 6 | The GUI will not accept a date of birth that is in the future. | Select one of the account type options (savings, checking, college checking, or money market) and name inputted in “FirstName LastName” format Set the date to some date in the future, then press continue. | Date of birth invalid. | Date of birth invalid. |
| 7 | The GUI will not proceed if an account selection is missing. | Have a valid name in “FirstName LastName” format and date of birth inputted where the date is before today’s date, then press continue. | Must select account type. | Must select account type. |
| 8 | The GUI will not proceed if the holder’s name is missing. | Select one of the account type options (savings, checking, college checking, or money market) and have a valid date of birth where the date of birth is before today’s date, then press continue. | Invalid name. Must be in format “FirstName LastName” | Invalid name. Must be in format “FirstName LastName” |
| 9 | The GUI will not proceed if the date of birth of the holder is missing. | Select one of the account type options (savings, checking, college checking, or money market) and name inputted in “FirstName LastName” format, then press continue. | Date of birth invalid. | Date of birth invalid. |
| 10 | The GUI shall proceed to the next step or action with valid inputs. | Select one of the account type options (savings, checking, college checking, or money market) and name inputted in “FirstName LastName” format, and have a date of birth where the date is before today’s date. | Able to proceed to the next pane (for opening, depositing, or withdrawing) or successfully close the account. | Able to proceed to the next pane (for opening, depositing, or withdrawing) or successfully close the account. |