|  |  |
| --- | --- |
| **Use Case Name:** | **1. Register a Scout** |
| **Description:** | |
| The Tree Lot Coordinator registers a Scout. | |
| **Preconditions:** | |
| 1. Person registering Scout is Tree Lot Coordinator 2. The Scout must have an existing record with Troop 209. | |
| **Workflow:** | |
| 1. Scout approaches the Tree Lot Coordinator and requests to register a Scout. 2. TLC requests the Scout’s identification information (including the following first name, last name, middle name, date of birth, phone number, email, and TROOPID). 3. Scout provides the identification information requested in step 2 to the TLC. 4. TLC validates the identification information provided in step 3. 5. TLC fails to retrieve a “Scout” record from the “Scout” folder using the TROOPID provided in step 3. 6. TLC creates a new “Scout” record containing the identification information provided in Step 2. 7. TLC files the “Scout” record in the “Scout” folder. 8. TLC informs the Scout the registration is complete. | |
| **Results:** | |
| 1. A Scout has been registered. | |
| **Alternates:** | |
| 1. The Scout already exists in “Scout” folder. 2. Scout does not provide all needed information in step 3 to the Tree Lot Coordinator. 3. Data type validation fails in step 4. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Scout” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **2. Update a Scout** |
| **Description:** | |
| A Tree Lot Coordinator updates the “Scout” record for a Scout. | |
| **Preconditions:** | |
| 1. The Scout must have an existing record with Troop 209. | |
| **Workflow:** | |
| 1. Scout requests the Tree Lot Coordinator to update a Scout. 2. TLC asks the Scout for information that can identify the Scout. 3. Scout provides the TLC with his/her identification information using which the Scout can be searched for (including one or more of ID, first name, last name, middle name, date of birth, phone number, email, and TROOPID). 4. TLC retrieves ALL “Scout” records from the “Scout” folder and creates a collection of all “Scout” records matching the search criteria provided in step 3. 5. TLC hands the Scout the collection of “Scout” records and asks them to select the desired record. 6. Scout selects the ID of desired record from the collection and sends it to TLC. 7. TLC uses the ID to retrieve the desired record from the collection created in Step 4. 8. TLC provides this “Scout” record to the Scout and requests them to make the desired changes. 9. Scout makes changes to the “Scout” record and hands it back to the TLC. 10. TLC validates the data in the updated “Scout” record sent in step 8. 11. TLC re-files the “Scout” record into the “Scout” folder. 12. TLC informs the Scout the update is complete. | |
| **Results:** | |
| 1. The Scout’s record has been modified appropriately. | |
| **Alternates:** | |
| 1. Scout does not know any of their identification information. 2. There are no matching “Scout” records. 3. Data type validation fails in step 10. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Scout” Folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **3. Remove a Scout** |
| **Description:** | |
| The Tree Lot Coordinator deactivates the “Scout” record of the requesting Scout. | |
| **Preconditions:** | |
| 1. The Scout has an existing record. | |
| **Workflow:** | |
| 1. Scout requests the Tree Lot Coordinator to deactivate a “Scout” record. 2. TLC asks the Scout for information that can identify the Scout. 3. Scout provides the TLC with his/her identification information which the Scout can be searched for with (including one or more of ID, first name, last name, middle name, date of birth, phone number, email, and TROOPID). 4. TLC retrieves ALL “Scout” records from the “Scout” folder and creates a collection of all “Scout” records matching the search criteria provided in step 3. 5. TLC hands the Scout the collection of “Scout” record and asks them to select a Scout. 6. Scout selects the ID of desired “Scout” record from the collection and sends it to TLC. 7. TLC uses this ID to retrieve the desired “Scout” record from the collection created in Step 4. 8. TLC sends this record to the Scout and asks for deletion confirmation. 9. Scout confirms this record is to be deleted to the TLC. 10. TLC changes the STATUS of the Scout’s record in the “Scout” folder to “Inactive”. 11. TLC updates the DATESTATUSUPDATED field of the Scout’s record to the current date. 12. TLC re-files the “Scout” record into the “Scout” folder. 13. TLC informs the Scout the removal was successful. | |
| **Results:** | |
| 1. Scout’s record is deactivated. | |
| **Alternates:** | |
| 1. Scout does not know any of their identification information. 2. Scout doesn’t confirm deletion. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Scout” Folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **4. Add a Tree** |
| **Description:** | |
| The Tree Lot Coordinator adds a Tree for a Scout | |
| **Preconditions:** | |
| 1. Person adding a Tree is the Tree Lot Coordinator. 2. Tree’s type exists in the database. | |
| **Workflow:** | |
| 1. Scout requests the Tree Lot Coordinator to add a Tree. 2. TLC requests all information about the Tree including the BARCODE and notes. 3. Scout provides the TLC with information about the Tree requested in step 2. 4. TLC validates all information given by the Scout. 5. TLC extracts the BARCODEPREFIX from the BARCODE provided in step 3. 6. TLC retrieves the “Tree Type” record from the “Tree Type” folder matching the BARCODEPREFIX from step 5. 7. TLC fails to retrieve a “Tree” record from the “Tree” folder using the BARCODE provided in Step 3. 8. TLC creates a new “Tree” record using all information about the Tree given by the Scout, the ID of the Tree Type retrieved in step 6 and inserts the current date into the “Tree” record. 9. TLC files the “Tree” record into the “Tree” folder. 10. TLC confirms to the Scout that the “Tree” record has been added. | |
| **Results:** | |
| 1. Tree has been added to the “Tree” folder. | |
| **Alternates:** | |
| 1. The Tree already exists in the “Tree” folder. 2. Data type validation fails in step 4. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Tree”/“Tree Type” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **5. Update a Tree** |
| **Description:** | |
| Modifies information of a Tree. | |
| **Preconditions:** | |
| 1. The Tree exists in the system. | |
| **Workflow:** | |
| 1. Scout requests the Tree Lot Coordinator to update a Tree. 2. TLC requests for the Scout to provide the BARCODE of the Tree. 3. Scout tells the TLC the BARCODE of the desired Tree. 4. TLC retrieves desired “Tree” record in the “Tree” folder using the BARCODE. 5. TLC hands the “Tree” record to the Scout and requests them to make the desired changes 6. Scout makes changes to the “Tree” record and hands back the “Tree” record to the TLC. 7. TLC validates the data in the “Tree” record provided in step 6. 8. TLC re-files the “Tree” record into the “Tree” folder. 9. TLC informs the Scout the update is complete. | |
| **Results:** | |
| 1. “Tree” record is updated with the desired changes. | |
| **Alternates:** | |
| 1. There are no matching “Tree” records. 2. Data type validation fails in step 7. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Tree” Folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **6. Remove a Tree** |
| **Description:** | |
| Removes the “Tree” record of the specified Tree. | |
| **Preconditions:** | |
| 1. Scout must have the BARCODE for the Tree. 2. Tree exists in the system. | |
| **Workflow:** | |
| 1. Scout informs the Tree Lot Coordinator they would like to remove a Tree. 2. TLC requests the desired Tree BARCODE from the Scout. 3. Scout provides the Tree BARCODE to the TLC. 4. TLC retrieves the “Tree” record from the “Tree” folder matching the BARCODE. 5. TLC hands the “Tree” record to the Scout for delete confirmation. 6. Scout confirms the “Tree” record in step 5 is to be deleted. 7. TLC confirms the STATUS field in the “Tree” record is not ‘Sold’. 8. TLC deletes this “Tree” record from the “Tree” folder. 9. TLC confirms to the Scout that the Tree has been removed successfully. | |
| **Results:** | |
| 1. A “Tree” record is successfully removed. | |
| **Alternates:** | |
| 1. Scout never confirms deletion. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Tree” Folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **7. Add a Tree Type** |
| **Description:** | |
| The Tree Lot Coordinator adds a Tree Type | |
| **Preconditions:** | |
| 1. Person adding Tree Type is the Tree Lot Coordinator | |
| **Workflow:** | |
| 1. Scout informs the Tree Lot Coordinator they would like to add a new Tree Type. 2. TLC coordinator requests type, description, COST, and BARCODEPREFIX of Tree. 3. Scout tells TLC the type, description, COST, and BARCODEPREFIX of Tree. 4. TLC fails to retrieve a “Tree Type” record using the BARCODEPREFIX provided in step 3. 5. TLC creates new “Tree Type” record of Tree Type based on type, description, COST, and BARCODEPREFIX of Tree. 6. TLC files the “Tree Type” record into the “Tree Type” folder 7. TLC informs the Scout that new Tree Type has been filed | |
| **Results:** | |
| 1. A new “Tree Type” record is added. | |
| **Alternates:** | |
| 1. The “Tree Type” already exists. 2. Scout does not provide all the needed information about the Tree Type. 3. Data type validation fails in step 4. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Tree Type” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **8.Update a Tree Type** |
| **Description:** | |
| Modifies the type of Tree provided to the Tree Lot Coordinator. | |
| **Preconditions:** | |
| 1. The Tree Type exists in the system. 2. Person updating Tree Type is Tree Lot Coordinator. | |
| **Workflow:** | |
| 1. Scout informs the Tree Lot Coordinator they would like to update a Tree Type. 2. TLC asks the Scout to provide the BARCODEPREFIX of the Tree Type. 3. Scout provides the BARCODEPREFIX for the Tree Type to the TLC. 4. TLC retrieves the desired “Tree Type” record using the BARCODEPREFIX provided by the Scout. 5. TLC hands the “Tree Type” record over to the Scout. 6. Scout makes changes to the “Tree Type” record and hands back the “Tree Type” record to the TLC. 7. TLC validates the “Tree Type” record. 8. TLC re-files the “Tree Type” record into the “Tree Type” folder. 9. The TLC confirms to the Scout that the Tree Type has been updated. | |
| **Results:** | |
| 1. A “Tree Type” record is updated. | |
| **Alternates:** | |
| 1. Data type validation fails in step 7. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Tree Type” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **9. Start a Shift** |
| **Description:** | |
| Starts a Shift for all Scouts. | |
| Preconditions: | |
| 1. Person starting the Shift is the Tree Lot Coordinator. 2. Scout has a Companion present. 3. All Scouts in the Session must be registered. | |
| **Workflow:** | |
| 1. The Scout requests the Tree Lot Coordinator to start a Shift. 2. The TLC requests the STARTDATE, START and STARTINGCASH cash from the Scout. 3. Scout provides the TLC with the STARTDATE, STARTTIME and STARTINGCASH. 4. The TLC validates the data provided in Step 3 by Scout. 5. The TLC creates a new “Session” record using the information provided in step 3. 6. The TLC files the “Session” record into the “Session” folder and notes the SESSIONID of the newly filed record. 7. The TLC retrieves ALL “Scout” records from the “Scout” folder and creates a collection of all these “Scout” records using all these “Scout” records. 8. The TLC hands this collection to the Scout and requests they select IDs of each Scout working this Shift. 9. For each SCOUTID the Scout selects:    1. The TLC asks the Scout for the STARTDATE, STARTTIME, COMPANIONNAME and COMPANIONHOURS for the Scout selected.    2. The Scout provides the TLC with the information requested in Step 9a.    3. The TLC validates the data provided in Step 9b.    4. The TLC creates a “Shift” record with the information provided in Step 9b, and the SESSIONID from Step 6.    5. The TLC files this “Shift” record into the “Shift” folder. 10. The TLC informs the Scout the Shift has been started. | |
| **Results:** | |
| 1. A Session is started. 2. Each Scout working is entered into a Shift. | |
| **Alternates:** | |
| 1. No information is given by Scout for one or more Shifts in step 9b. 2. Data type validation fails in step 4. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Scout”/“Session”/“Shift” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **10. End a Shift** |
| **Description:** | |
| Close a Session for all Scouts working. | |
| **Preconditions:** | |
| 1. A Session must be open. 2. The Scout has a companion present. | |
| **Workflow:** | |
| 1. The Scout requests the Tree Lot Coordinator to end a Shift. 2. The TLC retrieves the currently open “Session” record and notes the SESSIONID from this “Session” record. 3. The TLC retrieves all “Transaction” records in the “Transaction” folder that match the SESSIONID noted in step 2 and creates a collection of all these “Transaction” records. 4. For each of the “Transaction” records in this collection: 5. The TLC adds the TRANSACTIONAMOUNT to the TOTALCASHSALES value if the TRANSACTIONTYPE is ‘Cash’. 6. The TLC adds the TRANSACTIONAMOUNT to the TOTALCHECKSALES value if the TRANSACTIONTYPE is ‘Check’. 7. The TLC adds the STARTINGCASH amount from the open Session to the TOTALCASHSALES value to get the ENDINGCASH amount. 8. The TLC provides the ENDINGCASH and TOTALCHECKSALES amounts to the Scout and asks the Scout to confirm these values. 9. The Scout confirms these values to the TLC. 10. The TLC asks the Scout for any notes about the Shift. 11. The Scout provides the TLC with any notes about the Shift. 12. The TLC updates the “Session” record using ENDINGCASH and TOTALCHECKSALES confirmed in step 7, the current time as ENDTIME, and any notes provided in step 9. 13. The TLC re-files the “Session” record into the “Session” folder. 14. The TLC informs the Scout the Shift has been ended. | |
| **Results:** | |
| 1. The open Session has been closed. 2. Transactions have been totaled. | |
| **Alternates:** | |
| 1. The TLC fails to retrieve any “Transaction” records from the “Transaction” folder. 2. Scout does not confirm the ENDINGCASH and TOTALCHECKSALES amounts. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Session”/“Transaction” folder. | |

|  |  |
| --- | --- |
| **Use Case Name:** | **11. Sell a Tree** |
| **Description:** | |
| Scout sells a tree. | |
| **Preconditions:** | |
| 1. Tree must have ‘Available’ STATUS in the “Tree” folder. 2. Session must currently be open. 3. Scout needs the buyer’s contact information and method of payment. | |
| **Workflow:** | |
| 1. The Scout informs the Tree Lot Coordinator he/she would like to sell a Tree. 2. The TLC asks the Scout to provide the BARCODE of the Tree that he would like to sell. 3. The Scout provides the BARCODE of the Tree to the TLC. 4. The TLC extracts the BARCODEPREFIX from the BARCODE provided. 5. The TLC retrieves the “Tree Type” record from the “Tree Type” folder by using the BARCODEPREFIX computed in step 4. 6. The TLC extracts the COST value from the “Tree Type” record. 7. The TLC informs the Scout of the COST obtained in step 6. 8. The Scout informs the TLC of the new sale COST of the Tree. 9. The TLC asks the Scout to provide the name, phone number, email of the buyer and their method of payment. 10. The Scout provides the requested information to the TLC. 11. The TLC retrieves the currently open “Session” record and notes the SESSIONID from this “Session” record. 12. The TLC creates a new “Transaction” record and enters the buyer’s information, payment method, SESSIONID, TRANSACTIONAMOUNT, TRANSACTIONTYPE, the current date, and the current time. 13. The TLC updates DATESTATUSUPDATED to the current date in the “Transaction” record. 14. The TLC uses the BARCODE of the Tree to retrieve the “Tree” record from the “Tree” folder. 15. The TLC sets the STATUS of the Tree to ‘Sold’, and updates DATESTAUSUPDATED to the current date in the “Tree” record. 16. The TLC re-files the “Tree” record into the “Tree” folder. 17. The TLC files the “Transaction” record into the “Transaction” folder. 18. The TLC informs the Scout that the transaction has been filed. | |
| **Results:** | |
| 1. A Tree is marked as ‘Sold’ and the transaction is documented. | |
| **Alternates:** | |
| 1. The Scout doesn’t provide a sale COST for the Tree. | |
| **Entities Involved:** | |
| Scout, Tree Lot Coordinator, “Transaction”/“Tree”/“Tree Type” folder. | |

Tree Lot Coordinator (TLC) – Main Interface Agent

Scout – User

Loop – “For Each”

CRUD

Create - Validate, check for verification, create new record, file new record

Update – Search, create a collection, user selection from collection, update, refile

Delete – Same as an update, confirm selection, set status flag to inactive

BUSINESS LOGIC

Open A Shift – Scouts and an adult present with Scout, approach, TLC asks for shift info, Scout provides data model info, Session rolodex (new session record and file, note ID), ask info about each scout staffing shift, Create a collection of all scouts and Loop = for each scout the user selects get the appropriate ID and ask what times they start, and companion information, create a new shift record and file (End loop when done selecting), Inform user done

Closing a Shift – When closing a session, create collection of all transaction records whose session Id matches currently open session, sum cash in total cash sale, sum check in total check sale, note current date and time and note in session record, refile session record

Session = Main Shift

Shift = Scout in session

Selling a Tree – Get barcode from Scout, extract prefix, use prefix to get tree type record, get price, send price to user, user gives new sale price, Scout gives name phone number and email, transaction record from info given including the session id from the shift(precondition – A shift is currently open), file transaction record, retrieve tree record from barcode, set status to sold, refile tree record