CS MINI PROJECT

Prepared By Dhruv B. (191CH013) and Pranshu Shukla (191ME260)

Program Synopsis

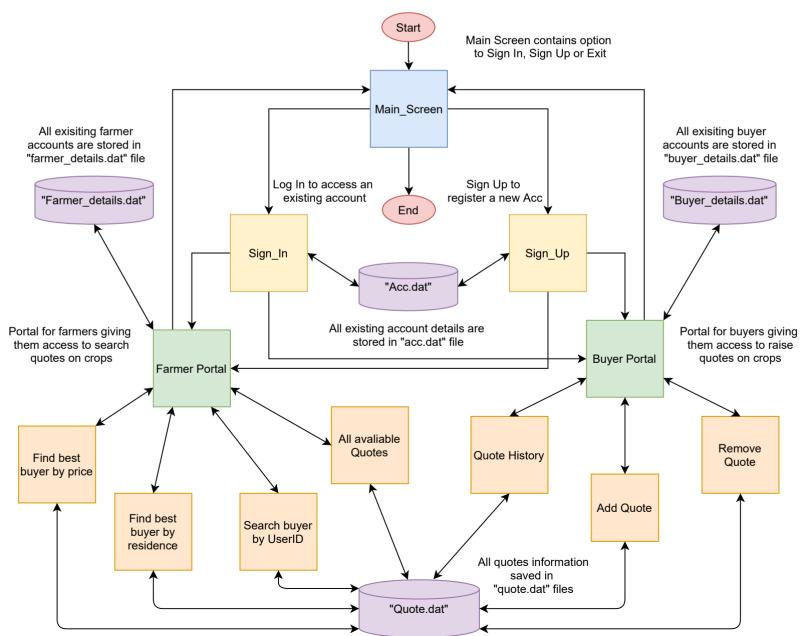
Farm-House is a user friendly program application that allows a quick and easy communication and transfer of information between a farmer and the buyer. Farm-House allows different users to choose which category they belong to (Farmer or Buyer) and provides them with appropriate options to make their interface as simple as possible.

Program Structure

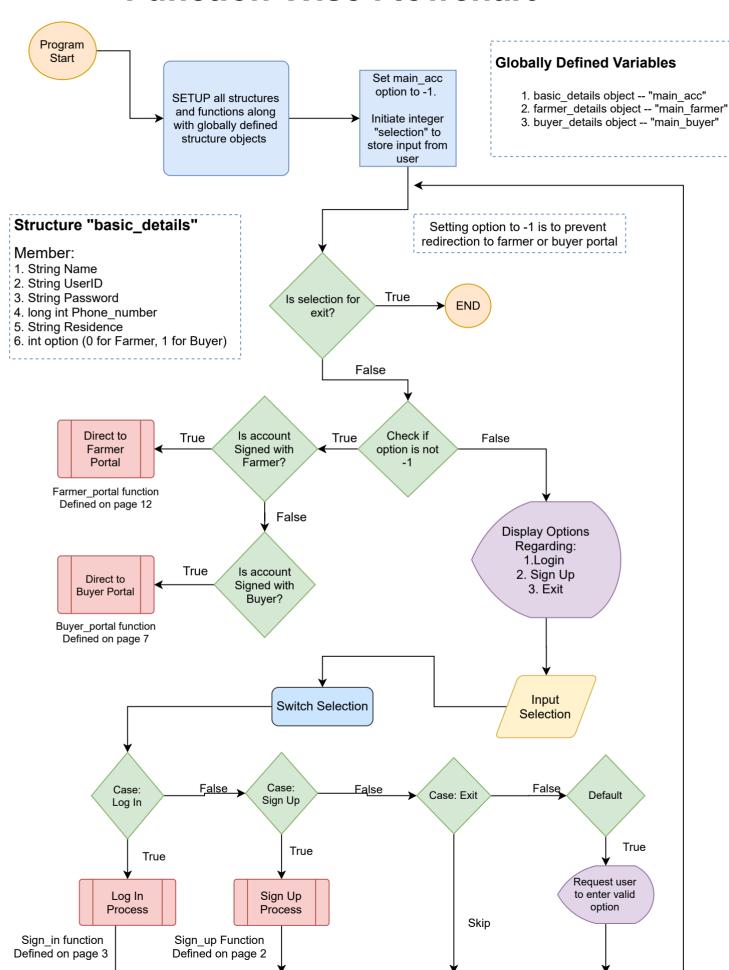
Farm-House uses 4 structures:

- 1. "basic_details" for storing UserID and Password of an acc as well as other details
- 2. "farmer_details" for storing farmer related information of the account
- 3. "buyer details" for storing buyer related information of the account
- 4. "buyer_quotes" for storing information on each quote issued by a buyer on the application

Program Overview Design

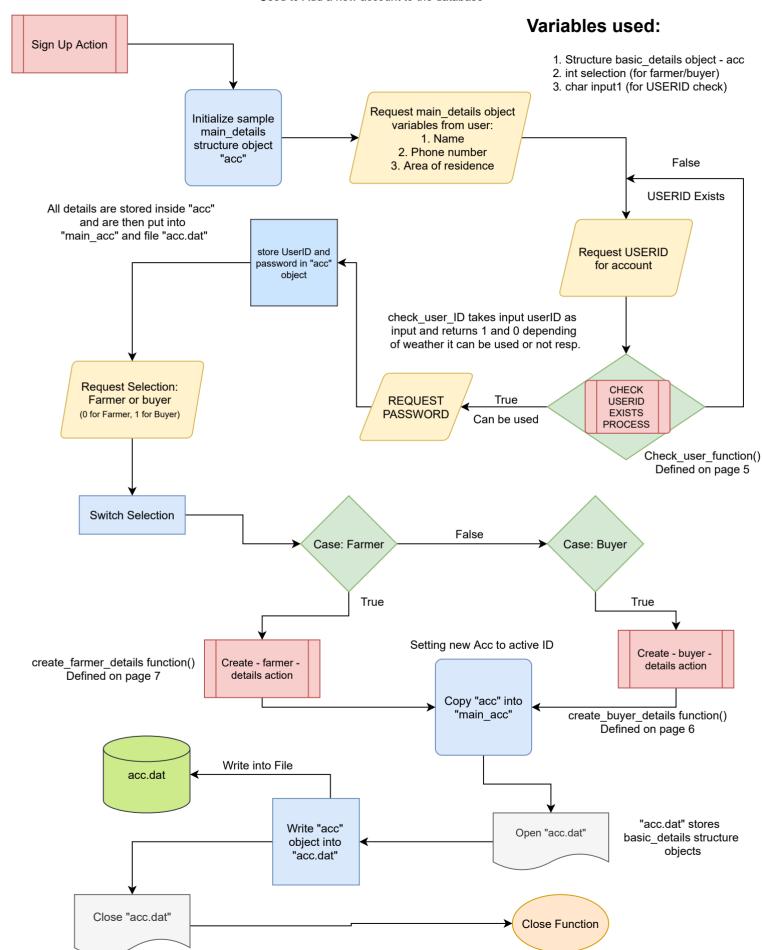


Function-Wise Flowchart



SIGN UP FUNCTION

Used to Add a new account to the database

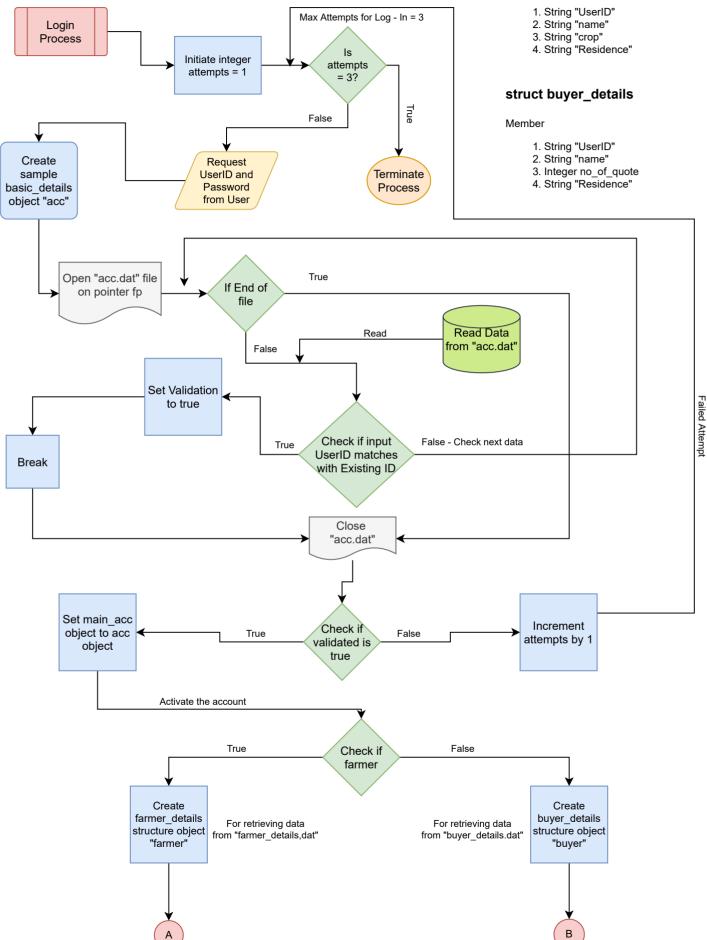


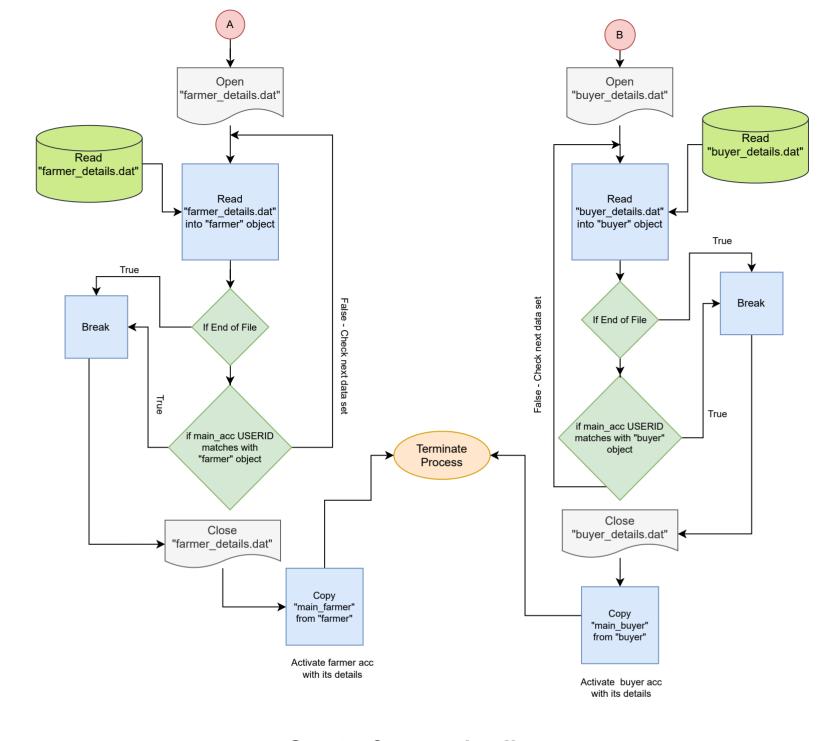
Sign In function

Used to Log-In into a new acc

struct farmer_details

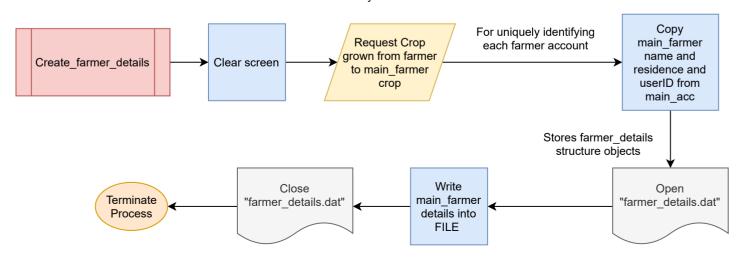






Create_farmer_details

Creates farmer details entry in file and sets to active account



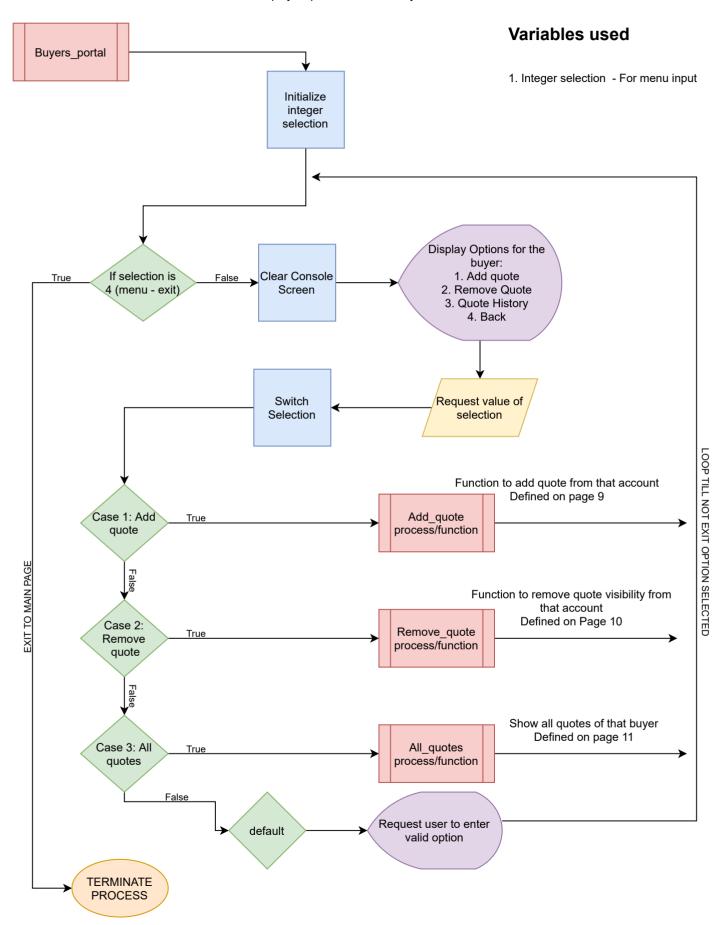
check_valid_userID

Variables used Used to check if the string the function is called is unique compared to other IDs 1. Basic details structure object "acc" - for reading data from file "acc.dat" Initialize Check Valid USERID object "acc" of Set Valid (Takes char string as 2. Integer "valid" - 1 for "is unique", to true structure input) 0 for "already used" basic_details Return variable: Open "acc.dat" 1 for "can be used/unique" Read acc.dat 0 for "is not unique" Read File False True check if structure into end of file acc object All accounts read Check if input char False matches with userID of Close "acc.dat" True Break **USERID** exists Set Valid to on other acc false Return valid create_buyer_details Creates new buyer account and makes it active. set Copy main buyer name and no of quotes residence and Create_buyer_details on userID from main_buyer as 0 main acc Open "buyer details.dat" Number of Quotes indicates For storing and uniquely the number of quotes of that identifying each buyer account specifically File for storing buyer details structure Terminate objects **Process** Write Close Write main_buyer "buyer_details.dat"

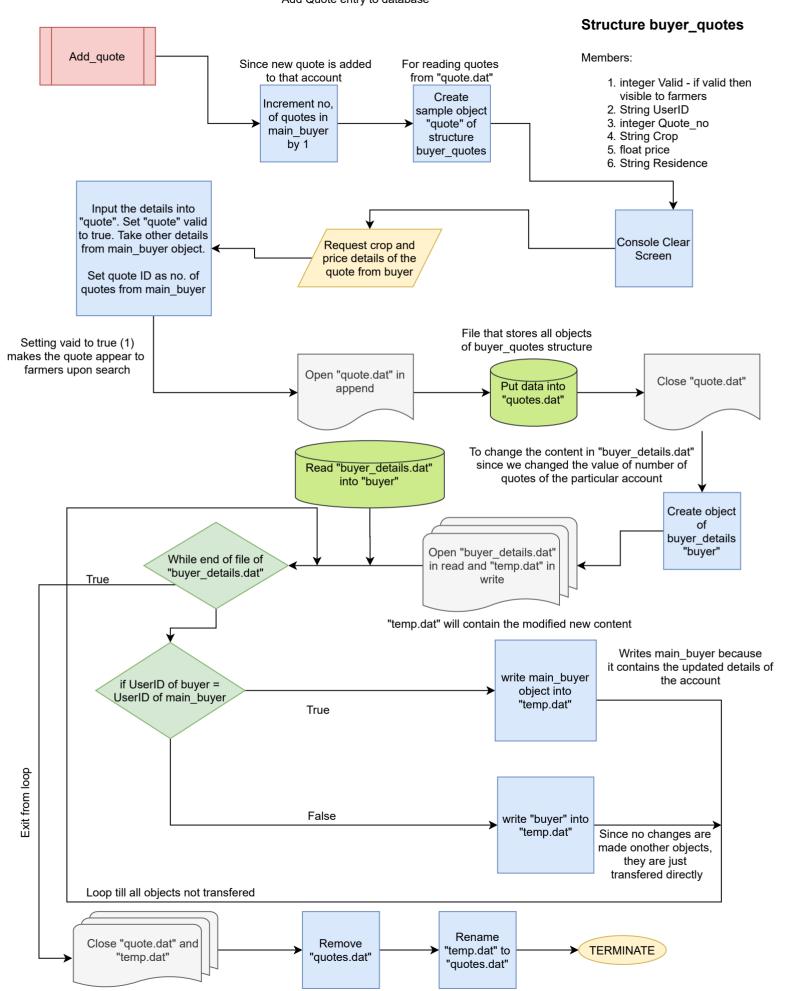
details into FILE

Buyer Portal

Displays Options for user if Buyer

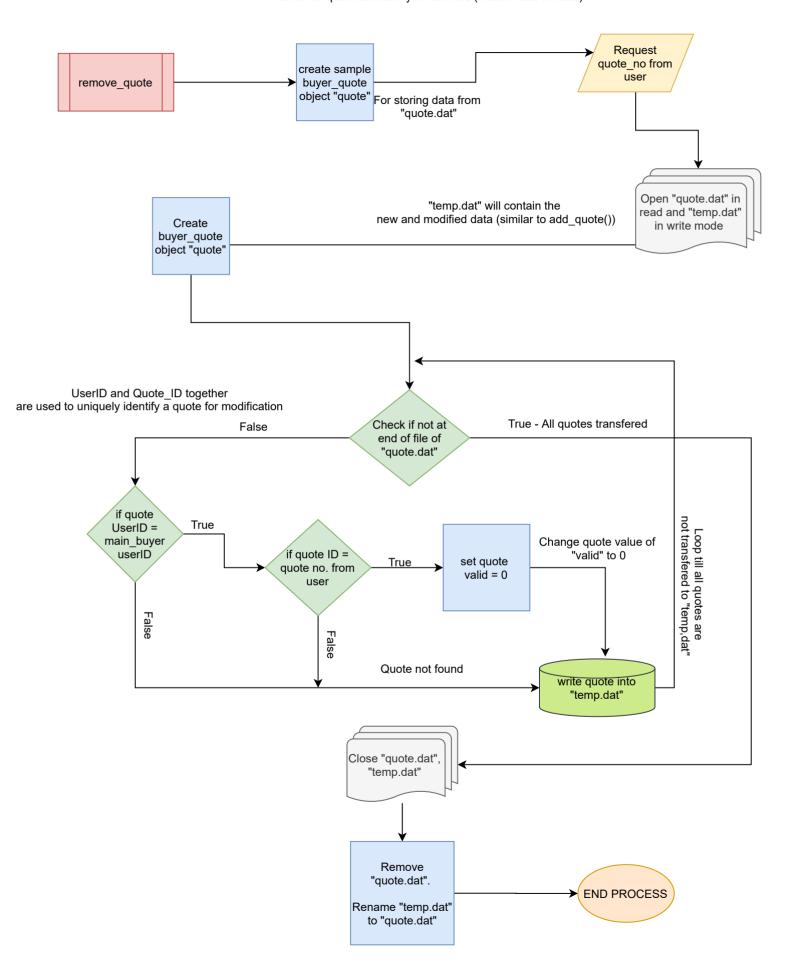


Add_quote() Add Quote entry to database

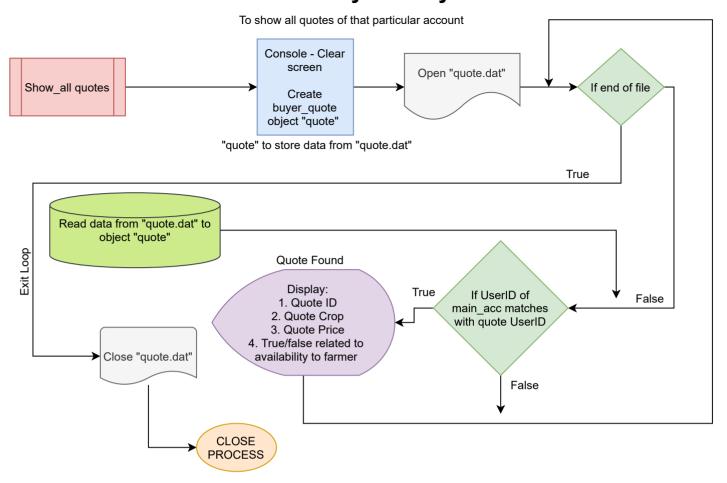


Remove_quote()

Removes quote availability for farmers (makes valid to false)

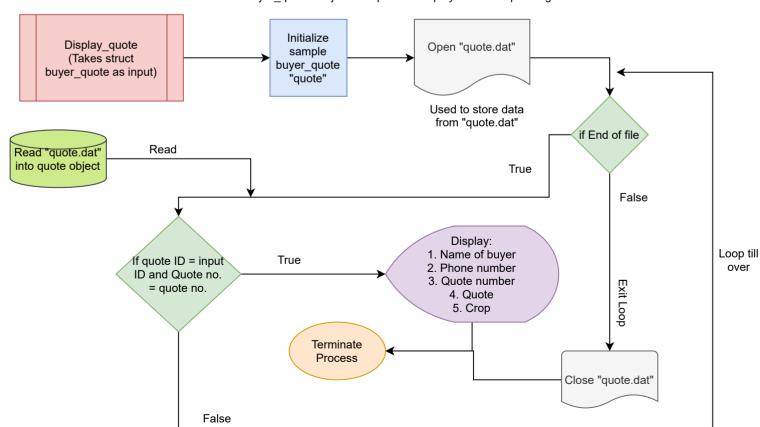


Quote History for Buyer



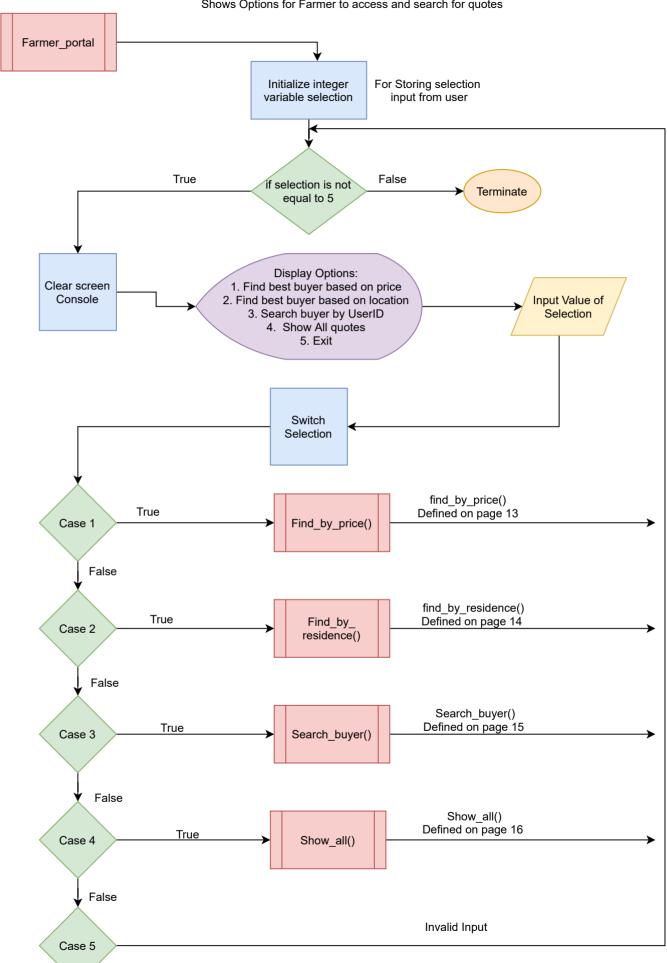
Display_quote()

Takes in structure buyer_quotes object as input and displays its corresponding details



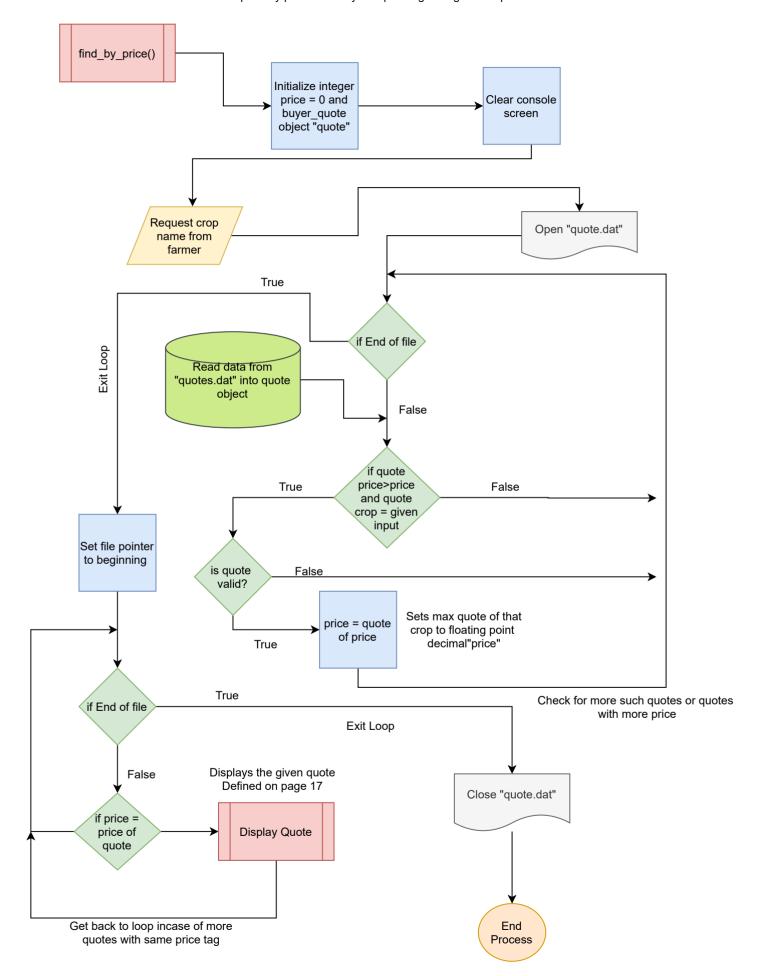
Farmer_portal()

Shows Options for Farmer to access and search for quotes



Find_by_price

Finds best quote by price for a buyer depending on a given crop



Find_by_location

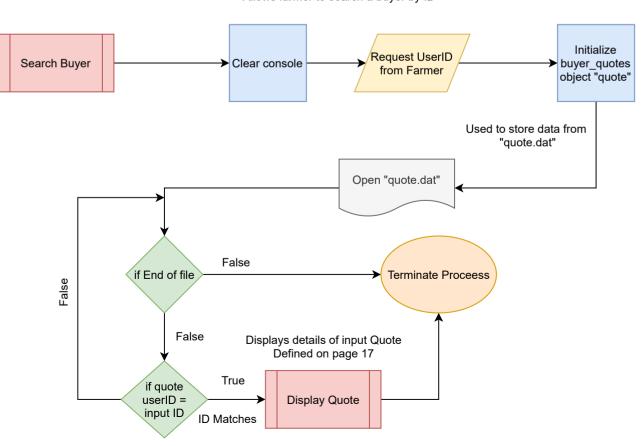
Used to find best quote available by location search Initialize Request user to buyer_quote Open "quote.dat" find_by_location input location and object "quote" crop area False True is quote True if End of file location = input is quote True location crop = input True is quote crop False valid? Display_quote() Defined on page False False Exit Loop Display quote Read "quote.dat" into quote object Quote Not match Quote Not match

Terminate process

Search_by_ID

Close "quote.dat"

Allows farmer to search a buyer by ID



Show_all_quote Shows all quotes that is accessible to farmers (Quote.valid is 1)

