

University of Louisville

ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

12-2006

Design of an interface for a decision support system for dairy collection.

Charles Milan Braidich 1983-
University of Louisville

Follow this and additional works at: <https://ir.library.louisville.edu/etd>

Recommended Citation

Braidich, Charles Milan 1983-, "Design of an interface for a decision support system for dairy collection." (2006). *Electronic Theses and Dissertations*. Paper 138.
<https://doi.org/10.18297/etd/138>

This Master's Thesis is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact thinkir@louisville.edu.

DESIGN OF AN INTERFACE FOR A DECISION SUPPORT SYSTEM FOR
DAIRY COLLECTION

By

Charles Milan Braidich
B.S., University of Louisville, 2005

A Thesis
Submitted to the Faculty of the
University of Louisville
J. B. Speed School of Engineering
in Partial Fulfillment of the Requirements
for the Degree

MASTER OF ENGINEERING

Department of Industrial Engineering

December 2006

DESIGN OF AN INTERFACE FOR A DECISION SUPPORT SYSTEM FOR
DAIRY COLLECTION

Submitted by: _____
Charles Milan Braidich

A Thesis Approved On

(Date)

by the Following Reading and Examination Committee:

Suraj Alexander, Thesis Director

Waldemar Karwowski

Dar-Jen Chang

ABSTRACT

DESIGN OF AN INTERFACE FOR A DECISION SUPPORT SYSTEM FOR DAIRY COLLECTION

Charles M. Braidich

December 2006

A decision support system (DSS) is a computer-based tool that extends an operator's cognitive abilities and makes him more effective in decision-making situations. This thesis discusses the characteristics and historical applications of decision support systems. It also discusses the contexts, operator requirements, and policies surrounding a DSS for dairy security; in this instance security includes both the physical security of the product and the integrity of the data concerning the product. Finally, this thesis discusses the methodology and resulting design of the DSS for dairy security. Specific tools and considerations involved in the design include the human centered design cycle (ISO 13407, 1999) and affordances of multiple types (Hartson, 2003). In addition to improving operator effectiveness, the Dairy Security DSS increases operator efficiency by reducing or elimination operator wait time due to the agitation process.

TABLE OF CONTENTS

	PAGE
ABSTRACT.....	iii
LIST OF FIGURES.....	v
I. INTRODUCTION.....	6
II. CURRENT SYSTEM DESCRIPTION.....	9
A. The Hauler’s Daily Routine.....	9
B. The Drop-Off Procedures.....	12
C. Opportunities for Improving Efficiency and Effectiveness.....	13
III. DECISION SUPPORT SYSTEM (DSS) EXAMPLES.....	15
A. Definitions and Descriptions.....	15
B. DSS Frameworks.....	19
C. Specific DSS Examples.....	21
IV. METHODOLOGY.....	24
V. PROPOSED SYSTEM.....	31
VI. RESULTS.....	37
VII. CONCLUSIONS AND RECOMMENDATIONS.....	40
REFERENCES.....	41
APPENDIX I. INTERFACE SCREENSHOTS.....	43
APPENDIX II. INTERFACE FLOW.....	59

CURRICULUM VITAE.....67

LIST OF FIGURES

FIGURE	PAGE
1. Milk Ticket Header.....	9
2. General milk hauler process flow.....	10
3. Individual Farm Information.....	11
4. DSS Framework (Bonczek et al.).....	20
5. Dairy Routing DSS (Butler et al.).....	22
6. The interdependence of human-center design activities (ISO, 1999).....	26
7. Collect Load Screen Shot.....	34

I. INTRODUCTION

A decision support system (DSS) is a computer based tool that assists human operators with processes that require both programmable logic decisions and non-programmable logic decisions. Programmable logic decisions are based upon perpetually true rules, such as if Condition A is true and Condition B is true then always perform Action One, or if the temperature in the freezer is above the selected temperature range then cool the freezer. A DSS is capable of making these decisions, so the human operator can focus on other system elements. Non-programmable logic decisions are those decisions where the outcome may vary depending upon a condition the DSS cannot know or cannot calculate; for example, if Conditions A and B are both true but Actions 1 and 2 are both still viable, and the correct decision depends upon Condition C, which is dependent upon the human operator's preference. A semi-real world example might be if a farmer's computer suggests that he cut hay in a field since the time is right, the weather will be good for several days, and the farmer's planner has him in town for several days, but the farmer may choose to ignore the suggestion if he has been suddenly called out of town or a blight has ruined his crop. While a DSS cannot autonomously make decisions of this type, it can assist the human operator in making these more difficult decisions.

The dairy industry requires a security system that protects both the physical product itself and the integrity of the data concerning the product during the collection process, where product is collected from individual farms and transported to processing plants. The best tool for securing this data, at multiple levels to be discussed later, is a DSS. The primary goal of this study is to explore the uses of DSS and design one to fit the specific needs of the dairy collection process. The objective of the DSS is to assist the operator in data collection and decision-making in such way that the data is accurate and relevant and that, in conjunction with other physical systems, the product is safe; the system must reduce or eliminate the likelihood of operator error, operator violation of procedure, and outside violation of the procedure and product. The secondary goal of this study is to make the entire system efficient for and attractive to all stakeholders, including government agencies, milk processors, milk marketing agencies, which connect farmers to haulers and processors, milk haulers, and dairy farmers.

Several approaches to the design of DSSs will be examined in the chapter titled "Decision Support Systems: Examples", and some of the features discussed in these examples will be incorporated into the design of the Dairy Security DSS. The human-centered design process (ISO 13407, 1999) will be employed to define an easy-to-learn, easy-to-use system that is relevant and will be utilized by the operators to support the new dairy collection process.

II. CURRENT SYSTEM DESCRIPTION

A. The Hauler's Daily Routine

The basics of a dairy hauler's work day are fairly consistent. First, the hauler picks up his truck and tank. These may be combined into the same vehicle and the vehicle may or not be at the hauler's residence. At this point the hauler checks his wash seals, tags from the last wash that must be broken to permit invasion of the truck's tank. The hauler also begins his milk ticket for the day, part of which is shown in Figure 1; the information entered here includes

4093		Dairy Farmers of America, Inc.		04075484		Raw Milk
Div	Hauler No.	Hauler Name		Ticket # on page 1		
	463558	BIG BLUE				
Load No.	Route No.	Pickup Date (MMDDYY)	Tanker No.	Pickup Truck No.		Total Number of Pages
	2808	071905	k88	100		1 of 1
Handler Name	Handler No.	Delivery Date (MMDDYY)	Product			

FIGURE 1 – Milk Ticket Header

ticket number (already printed on form), milk marketing agency and number (may also be already printed on form), hauling company name and number, product type, pick-up date, tank number, truck number, and, now or at route's end, driver's license / permit number.

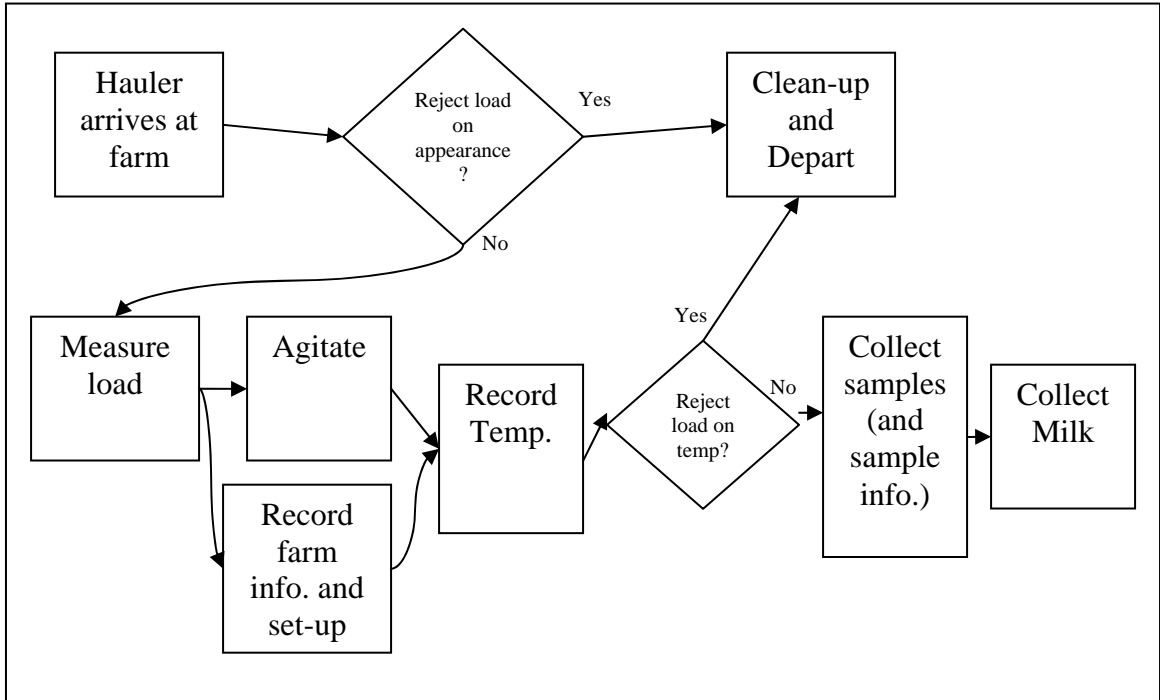


FIGURE 2 – General milk hauler process flow

The hauler then embarks upon his route, the general flow of which is shown in Figure 2. Arriving at the first farm the hauler connects his milk hose to his pump and slides the other end into the port in the milk house wall; he also connects his pump to the electrical outlet (if one is available). The hauler then enters the milk house, inspects the milk for visual or olfactory cues (he may reject the milk at this time if the milk is red (with blood) or smell sour), and measures and records the depth of the milk in the tanks. He then starts the agitator, a device, generally electrical, that stirs the milk to ensure homogeneity. While the milk is agitating, a process that lasts five minutes for the smallest tanks, the hauler converts the depth reading to a weight, which he records, and connects his hose to the farm tank's output valve. During this time he also applies labels

to the sample containers and his milk ticket; in addition to a barcode, the labels may also have farm and hauler information, such as farm name and number, hauler name and number, and date. Finally, the hauler uses this time to fill in his milk ticket with the information that is currently available, including farm number, pick-up time and date, bulk tank unit (the farm tanks' group identification number), the tank number (if more than one are present on this farm, farmer name, and product grade; see the individual farm section from the milk ticket found below in Figure 3. These four steps, hose connection, conversion, labeling, and data entry, take an

Div	Producer No./Handler				Weight	Military Time (HEMM)	Temp	Stick Reading
	5	5	7	8	8	8		
Payee	Tank	Pickup Date (MMDD)	F/R		7	9	0	5
	1	0	7	1	9	0	8	3
			<input type="radio"/> F <input type="radio"/> R					
					21-900	A		28-1
					Jones			JH

FIGURE 3 – Individual Farm Information

experienced driver no more than one minute. This idle time represents an opportunity that will be discussed later in this study.

Once agitation is complete, the hauler takes the milk's temperature. If the temperature is acceptable the hauler takes the require number of samples, which he stores in an ice bath in a cooler on his truck, opens the output valve on the farm tank and flips a switch to power his pump to begin the pumping process; meanwhile, he records the temperature, signs the milk ticket for that individual pickup, and fills out the milk house sheet. The hauler often leaves the agitator running until the milk level in the farm tank drops below the agitator's reach.

When the tank has been emptied the hauler turns off the power to his pump and disconnects his hose. He then hoses out the tank to remove any residual milk that may be present; once the rinsing is complete, the hauler connects the cleaning system to the tank (to several portals in the top of the tank and the output valve). The hauler initiates the cleaning cycle, hoses off the milk house floor, stores his equipment in his truck, and leaves. The entire cleanup process takes less than five minutes in most cases. The steps found in this paragraph and the previous paragraph are repeated at each farm the hauler services during the day. Any driver changeovers are recorded; this includes date and time. The original driver must sign the bottom of the milk ticket, approving all pickups up to this time.

B. The Drop-Off Procedure

When the driver's route is over or his tank is full, he heads to the milk processor, which may be one to three hours away. Upon arriving at the receiving station, the hauler meets with a receiver. The hauler signs the milk ticket, and the receiver enters his number, the delivery date and time, and his signature. The receiver sends the final load sample, which the hauler took at the end of the route, to the testing room and the farm samples to the holding refrigerator for later testing; meanwhile, he inspects the tank, inspects the seals, which must match the wash tag, inspects the load for appearance and odor, and checks the load's temperature.

Upon approval of the load's antibiotic test result, the load is pumped into one of the plants silos. At any point before this moment, the receiver would have rejected the load if any of his inspections were failed. Rejected loads are resealed and sent out; the hauler, in combination with the hauling company, milk marketing agency, and the farmer, determine how to handle the rejected milk. Empty tanks, whose loads were pumped into the processor's silos, are usually washed at this point; washing may be foregone for several reasons, such as a second route or a different preferred washing station. The receiver handles the washing, locking, and resealing of the truck tank, and the related documentation.

If instead of a processing plant the hauler delivered his load to a semi-tanker at a pumpover station, the procedure would be much shorter. The receiving hauler may wish to inspect the load if he is present. If the pumpover is partial (only part of the hauler's load is being pumped onto the larger truck) antibiotic testing is mandatory, otherwise it is optional. Samples are given to the receiving hauler or placed in an on-site refrigerator. The pumped-over volume is recorded, and a copy of the milk ticket is given to the receiving hauler. It should be noted that the receiving hauler may not be present at the time of pumpover. The hauler may now wash his tank if he is empty.

C. Opportunities For Improving Efficiency and Effectiveness

There are several opportunities to improve both operator efficiency, which, as will later be discussed, is not within the domain of a DSS, and operator effectiveness, which is within that domain. The apparent opportunity for

efficiency improvement, discovered while observing the milk hauler's routine, is the time he spends waiting while the agitator runs. As previously mentioned, the tasks he performs during agitation rarely take over one minute, while the agitation time is proportional to the size of the tank and takes approximately five minutes for a small tank and approximately ten minutes for a medium tank.

There exist several opportunities for improving the hauler's effectiveness, namely error reduction and violation reduction. Data entry errors (or violations, in the case of temperature) are incredibly easy to make, especially when the hauler must make look up values for depth-to-weight conversion or read conventional thermometers; those opportunities for data entry error should be removed.

Directly related to this are translation error and interpretation errors; when using paper to record this much data the hauler can transpose figures to incorrect positions on the milk ticket, and, in addition, the receiver or marketing agency may be unable to read the hauler's writing. Also, if the temperature of the milk samples exceeded a certain value, generally 42 degrees Fahrenheit, it could throw off the reading of many tests performed on the milk, including butterfat content. In addition to these data errors, which may affect the income of the processor, the agency, the hauler, and several farmers, there are other errors and violations which may affect the safety of the consumer. One such error is if the hauler accidentally began the sampling and pumpover process too soon, before the agitation process was complete; this may also be a violation if done consciously; the possibility or temptation to commit this error/violation must be reduced or eliminated.

DECISION SUPPORT SYSTEM (DSS) EXAMPLE

A. Definitions And Descriptions

As mentioned earlier, DSS's purpose is to aid human decision-making in semi-structured tasks where analytical tools are of value but where human judgment is still essential (Fiallos, 12); they require two-way human/program interaction and are meant to improve the human's effectiveness not efficiency in these semi-structured tasks. DSS were introduced in the latter quarter of the Twentieth Century; as computer power and availability increased DSS received more attention, and DSS usage shifted from only managers to all variety of operators making time-crucial and/or difficult decisions. However, little has been done to structure their design; this is likely due to the range of DSS applications and the fast-paced revolutions experienced by computer technology during this period.

Keen (1983) says that DSS represent one school of thought on the role of computers in the management decision-making process. That role being to:

1. Assist managers in their decision process in semistructured tasks;
2. Support, rather than replace, human judgment; and
3. Improve the effectiveness of decision making rather than its efficiency.

All of these describe the role the dairy security DSS should perform. According

to Keen, the effectiveness of the design relies on the programmer's or designer's understanding of the management decision making process and the standards for computer-based decision

aids. Keen and Scott-Morton (1978) performed a study that found no typical design for a DSS.

Alter et al. (1980) said that DSS should focus on the organization and emphasize the following:

1. Decision focus, aimed at top managers and executive decision makers.
2. Emphasis on flexibility, adaptability, and quick response.
3. User initiation and control.
4. Support for the personal decision making styles of individual managers.

The first characteristic is relevant (to a point); a DSS's goal should be a decision, such as accepting or rejecting milk, but this applies to even low-level operators.

The fourth characteristic might apply less in more structured environments, like milk collection where any system would be partially aimed at data collection.

Geoffrion (1985) stated several principal characteristics of a DSS that differentiate it from a merging of the Management Science and the Operations Research school of thought. These characteristics are less constrained than, or at least modernized versions of, older characteristics that take into account more modern technology and business environments. These characteristics will be utilized in this study's design process, which is described in the later chapters.

1. It often adopts a multiple-pass approach to system development rather than the more traditional serial approach of requirements, specifications, detailed design, programming, testing, and implementation.
2. It places a high value on flexibility of system use and adaptability to changing user needs.
3. It strives for a specific integration of data sources and models.
4. It puts users first and underlying technology second.

Mittra (1986) described situations where DSS are applicable.

1. Existence of large database.
2. Necessity of computation in the process of arriving at a solution.
3. Existence of some time pressure for the final answer.
4. Necessity of upon available alternatives by asking, for example, “what if” questions.

Mittra’s first characteristic applies to the dairy security study since the vast number of statistics, the location for each action, and more that must be recorded nationwide each day will create a very large database. A couple of examples of how computation and “what if” characteristics are combined to apply to this study are

1. There is a limit, usually 72 hours, on the maximum time allowed between the last wash and the pick-up time; the DSS designed in this study should help haulers adhere to this law.

2. What if the load size will breach the truck tank's capacity and the hauler forgets or miscalculates? The DSS for dairy security can assist in avoiding a large mishap.

Mittra defined a DSS as a system that provides all relevant data in an easily usable form to improve the effectiveness of a manager's decision-making process; the manager would formulate the problem using the computer system's interactive and probably menu-driven front end.

Mittra's definition is similar to that of J. Little (1970): a DSS is a "model-based set of procedures for processing data and judgments to assist a manager in his decision making." Little's characteristics of a successful DSS can be found buried in the aforementioned Geoffrion's characteristics; Little said that such a system must be simple, robust, easy to control, adaptive, complete on important issues, and easy to communicate with. These evolutions on Little's principles point to Little as one of the forefathers of DSS.

Ginsberg and Stohr (1982) said DSS improve managerial decision making in unstructured tasks by extending a manager's cognitive capabilities, leaving him free to exercise his judgment where it is needed. While the first part of this definition does not completely apply to the subject of this study, the second part is exactly the sort of cognitive relief the system should provide the hauler by, for example, reminding him of the capacity limits of his truck should a proposed farm pickup exceed that capacity.

Klein, Konsynski, and Beck (1985) again offer a slightly off target, slightly on target definition, with respect to the proposed system: a DSS "is and

interactive process that helps a decision maker utilize data and models,” which is exactly what the dairy DSS should do, but the dairy DSS will apply to a relatively standard group of problems, whereas Klein, Konsynski, and Beck only had “new and relatively unstructured” problems in mind.

B. DSS Frameworks

There are several frameworks for DSS that should be discussed when proposing a new DSS design. R. Bonczek et al. (1981) described a generic three-part DSS framework, the components of which are a Language System, a Knowledge System, and a Problem Processing System. The Language System provides linguistic facilities; it helps the user state the problem to be solved. The Knowledge System contains a body of knowledge concerning the problem domain. The Problem Processing System is the only dynamic component of the three-part DSS system; it acts as the mechanism that accepts the problem from the representative Language System and uses the knowledge represented in the Knowledge System for decision support. The Language System varies only in the level at which it can describe the problem and the formality of its procedure; it is characterized by the syntax, or statements, commands, or expressions, it affords the decision maker. The Knowledge System varies in the knowledge it contains and its organization; this vast volume of facts and relationships is something the user has neither the time nor urge to commit to his own memory.

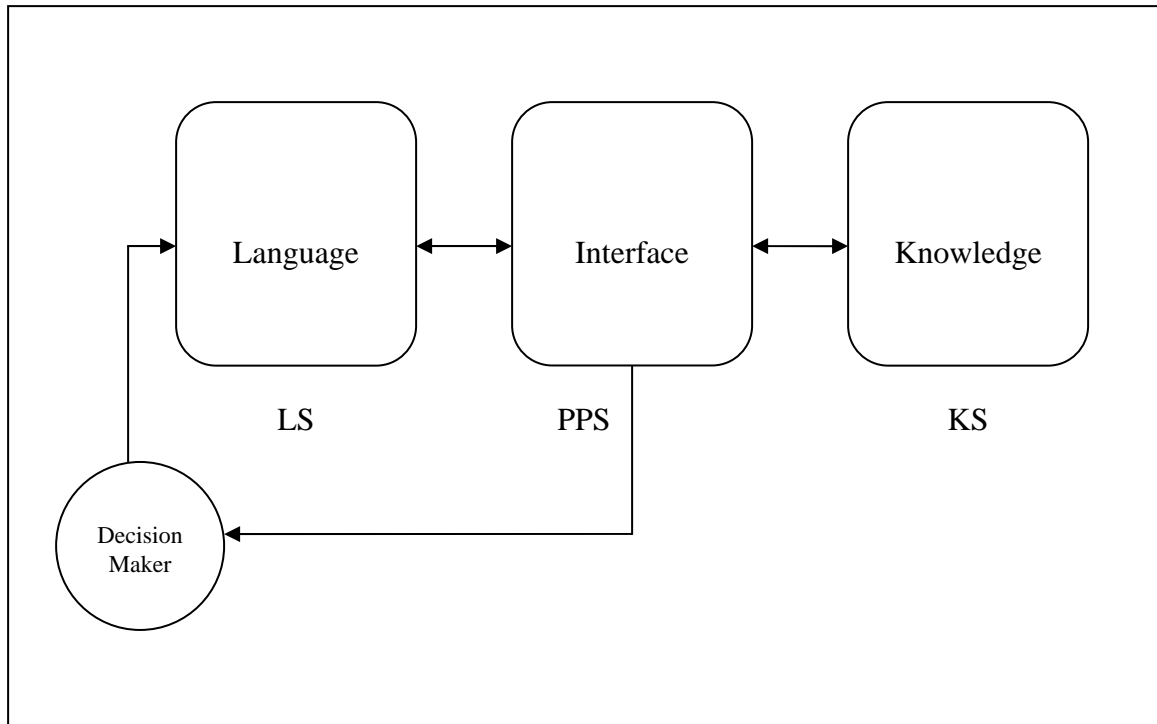


FIGURE 4 – DSS Framework (Bonczek et al.)

The Problem Processing System must be able to recognize problems types and transform that problem into an executable plan. It must then carry out that plan and develop an answer. When models are involved, which this study finds to be the common case, the PPS must be capable of interfacing models with data in order to generate some facts or expectations.

The DSS described by Bonczek et al. uses information about a given problem domain (KS) and applies it to a problem described by the user (LS) using a certain process or set of algorithms (PPS).

Geoffrion et al. (1983) described a similar three-part framework. His DSS supports decision-making using three major subsystems: a database, a model base, and a decision maker. The database is a store of knowledge and the software that manages it. The model base contains strategic, tactical, and

operational models and sub-models from which new models may be created. The final component is the man-machine interface, the decision maker, which is mainly the user, the terminal, and the command language. House makes no mention of the program for matching the user's problem description to certain models as a specific system component.

C. Specific DSS Examples

One specific DSS designed by Baxter, Gordon, Monk, Tan, Dear, and Newell (2005) was actually the second attempt to introduce a DSS to a neonatal intensive care unit (NICU). In the NICU many key parts of a baby's development take place outside the womb; the most common treatment for patients is mechanically assisted ventilation. Controlling mechanically assisted ventilation can be a difficult task, a task made even more difficult when there are multiple patients and several shifts of medical personnel that must hand-off duties each day.

The previously introduced DSS was ESNIC; it required manual input of data and did not work within the context of the NICU. When the new system, FLORENCE, was designed the capability now existed to get data electronically from intra-arterial sensors and ventilators. In addition to eliminating the manual input of data, the designers also charged themselves with minimizing false alarms, since alarms sound in the NICU "fairly frequently", making the DSS alarms unique and noticeable, and offering recommended courses of action with

explanations. By the same token it also required explanations from medical staff if they chose a non-recommended course of action.

One lesson learned in the NICU DSS example is that communication and feedback are key; both of these must be clear between different organizational levels and across time. Also, the study recognized implications that go beyond calculations, like avoiding false alarms and making emergency alarms noticeable.

In 2005, Butler, Herlihy, and Keenan proposed, in theory, a DSS for managing milk collection that had a particularly logistical bent. Their system focused on routing trucks to different farms and processors to provide milk to processors at the least cost. Their system's data flow is modeled below.

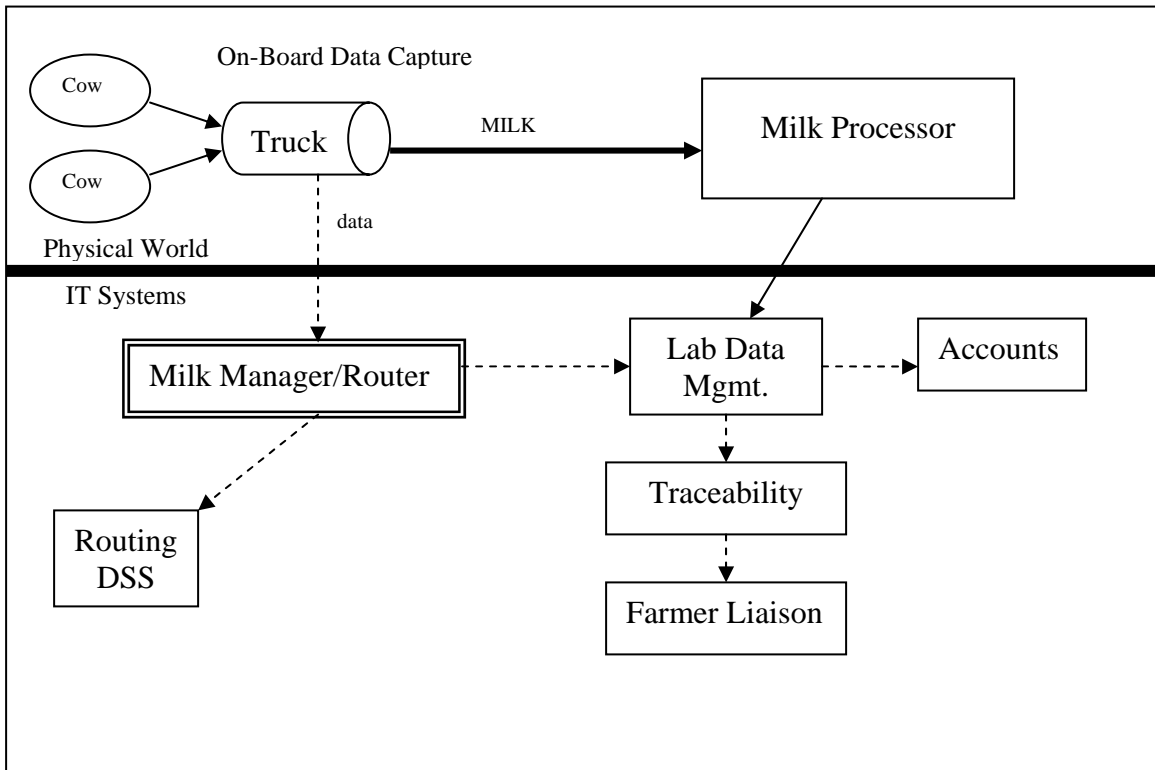


FIGURE 5 – Dairy Routing DSS (Butler et al.)

The general DSS model proposed by Butler et al. is not at odds with the objectives of this study; on the contrary, it may be the ultimate product of this study. However, this study is the first step in that direction; it will be composed of the truck, on board data capture, and the data sent to a the higher authorities listed as stakeholders for this study, government agencies, milk marketing agencies, processors, and farmers, with each source receiving data that is important to them and that they are authorized to obtain. Traceability to the farm and cow level, which will affect both farmers and processors monetarily and will involve the liaisons that work between industry representatives and the government, may be the next step. After that building block is complete, optimizing truck flow through logistics management would complete the system this study begins and parallel the system proposed by Butler et al..

METHODOLOGY

The major framework used in generation of this system is human-centered design processes found in ISO Standard 13407 Human-centered design processes for interactive systems (1999). The rationale for employing human-centered design is quite powerful, the essence is that users are more likely to use it because it will make their lives easier, and it is beneficial to the organization because a product of higher quality will be generated at a lower support and training cost. The basic process for human centered design can be found in the diagram below. This iterative cycle matches Geoffrion's first design characteristic, which calls for a multi-pass approach to system development.

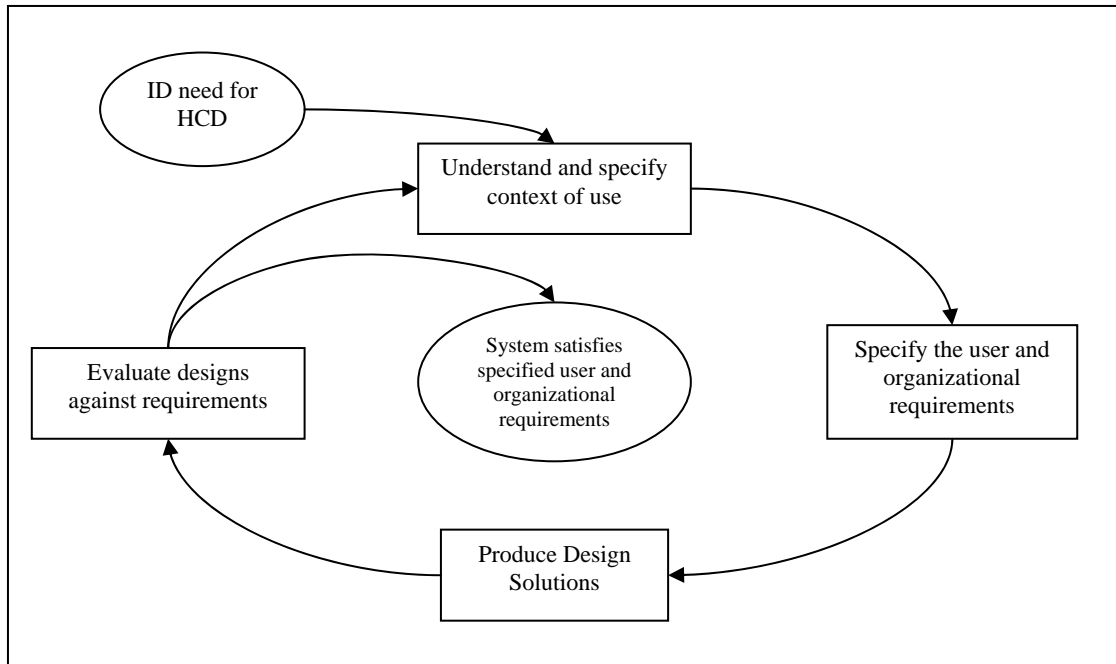


FIGURE 6 - The interdependence of human-center design activities (ISO, 1999)

This process is followed in the design of the proposed system, at first through conscious practice but then as the logical, natural course of action. Below is a summary recounting of how the HCD design process has been followed in this study. Other components of human-centered may be employed in the design of the decision support system but may or may not be detailed in this document.

The need for HCD was identified as soon as it was known that there exists a need for a human-machine interface. At the first meeting for the Dairy Security Project the daily routine of the dairy hauler was discussed; from this the most basic functional requirements for the system could be outlined. At the second meeting, the project team met with industry collaborators, including haulers, milk marketing agencies, and milk processors; those collaborators aired their concerns and needs, adding substantial flesh to both the context and the user

and organizational requirements outlined earlier. At the same meeting a very rough design was pitched to those stakeholders, and their reviews and opinions were taken into account.

After that meeting, a day was spent conducting observations on the actual collection process. This provided a better understanding of the context to the design team; it also pointed out areas of interest with respect to prospective efficiency savings. At the third and most recent project team meeting, a more specific design was discussed and recommendations from both haulers and government representatives helped shape the majority of the final design proposed in this study. In addition to that third meeting, several versions of this design have undergone mild revisions after the discussions with dairy policy experts from industry regulators.

After this design has been submitted to the system builders it is likely that the system they build will be put before the industry collaborators, and their recommendations, given their deep understanding of the contexts in which the system will be used and the demands they will place upon the system and vice versa, will spark at least one more cycle of the HCD cycle. That would be representative of a genuine human-centered design, and display the design team's commitment to a truly usable system.

In addition to following the HCD design cycle, other tools must be employed to create the most usable system possible. Hartson (2003) discusses the presence and needs for more affordances in modern society and the need for a universal terminology for use among designers of systems and tools for use by

people other than themselves. Affordance is a term coined by Norman in Design of Everyday Things (1988), and he defined it as a characteristic of a device or interface that allows its operation; since then many other experts in cognitive task analysis and product design have expanded on the subject, including Gaver (1991), who saw affordances “as a way of focusing on strengths and weaknesses of technologies with respect to the possibilities they offer people”. Below is a table of the different types of affordances, a common example of each, and an example of each within the proposed system.

TABLE I
AFFORDANCE TYPES, DEFINITIONS, AND EXAMPLES

Affordance Type	Description	Common Example	Proposed System Example
Cognitive Affordance	Design feature that helps users in knowing something	A button that helps users know what will happen if they click on it	The taskbar will display of battery silhouette filled to a level indicating device’s battery level (90% full equals a 90% full battery)
Physical Affordance	Design feature that helps users in doing a physical action in the interface	A button that is large enough so that users can click on it accurately	Buttons that are large enough and consistently positioned to allow easy use, even in extreme conditions
Sensory Affordance	Design feature that helps users sense something	A label font size large enough to read easily	The battery silhouette will contain green color if battery level in upper third; yellow in middle third; and red in last third
Functional Affordance	Design feature that helps users accomplish work	The internal system ability to sort a series of numbers	Auto-populating last used User ID in Sign-In Screen to forego re-entry via alpha-num. keyboard on screen

According to Torres (2002), when designing handheld user interfaces provides additional challenges since the screen size is often much smaller than desktop units, and instead of a mouse and keyboard the user only has a stylus to manipulate the interface. Torres challenges the designer with these two tasks:

1. Simplifying the demands on the user for input and interaction
2. Exploiting the constrained display area

Examples of satisfying the first demand are keeping many responses in lists to eliminate manual user entry and scanning barcodes (or other forms of semi-automatic location and equipment identification).

Another guideline that bears attention can be found in Spolsky's User Interface Design for Programmers. A designer should strive for consistency, not only within their own product but also with respect to other user interfaces with which the operator is most likely familiar. For example, Visio is not originally a Microsoft product, but the creators of Visio designed it in such a way that many Windows and Microsoft metaphors remained consistent throughout their application; therefore new users learned to use Visio with relative ease.

Evidence of the significant impact of the information this study has found on interface and system design, especially with respect to usability, can be found in the screen descriptions and screen flows of the proposed system design. To support these two design documents, the following list of usability guidelines for use on this specific system are:

1. Fields that can and should be tapped to enter information should be raised like a button.

2. Buttons that are not click-able should be visible but obviously unusable. Make them apparently buttons but gray-out the text.
3. Make text easily readable; use Arial font of a size that fits the chosen handheld device's screen.
4. Buttons and click-able fields should be large (not huge); our users will be in a hurry and perhaps not extremely dexterous with respect to a stylus.
5. Orient data entry around objects on the screen, not commands. For example, if there is a field for my name I should be able to tap it with my stylus and a qwerty-style keyboard should appear on screen. The keyboard should be large enough to be usable. In addition, the user should be able to tap a key and switch to a number pad (and back and forth between the two); the key's use should be apparent. (Make it "abc/#".)
6. When the tapped field is primarily a number field the number pad, not the keyboard, should appear first; they may choose to switch to the keyboard (and back and forth between the two).
7. Messages at the bottom of the screen should be discernable from normal screen text. Change the color place a line between the screen text and the message.
8. The task bar at the bottom of the screen should be obviously that, and the icons on it should be understandable. Colors are on the screen descriptions page, but the shapes should be:

- a. Power – an I/O switch
 - b. Lock – a padlock
 - c. Battery Level – the shape of a battery
 - d. Tank Level – sideways cylinder or truck
9. Some fields (e.g. partial pick-up fields, scan tractor, and destination truck) are only valid when other fields are of a certain value. If a field is not valid, then it should be invisible.
10. Keep button location (Accept, Reject, Cancel) consistent.
11. Replace buttons with verbage with buttons with icons where possible
- a. Good examples: replace “Accept” with a checkmark; replace “Cancel” with an “X”
 - b. Bad examples: “Reject” may be difficult to replace with an icon

By adhering to the design documents, following these guidelines, and continuing the iterative cycle of the HCD design process, the system builders will create a highly usable and functional user interface and DSS.

PROPOSED SYSTEM

The previous chapters of this thesis have discussed the reasons why a DSS is the correct solution to the dairy industry's problem with respect to information security and hauler effectiveness and the theories and reasoning that have gone into designing this DSS. This chapter will discuss very briefly the proposed system design and some of the points at which principles discussed in the DSS Examples and Methodology chapters have been applied. Let this point be clarified: This design is for the logic behind and appearance of the handheld interface only. The overall system will be composed of many of these units, truck-based computers, and central servers and databases; the interaction of these many components is to be designed by other members of the project team charged with building the physical systems.

When the user signs-in, either to unlock a locked screen or because the device was just turned on, the program validates the sign-in, and, if it is the first sign-in of the day requests several pieces of information like truck number, tractor number (if applicable), cargo. It also associates that information and the hauler, hauling company, and milk marketing agency information with today's load. Additionally, if the truck has not been washed within the 72 hours, the

system requires the user to either lock the device or enter and go through the wash screen. This ability to ask, “What if the truck tank has not been washed in 72 hours?” display Mitra’s fourth characteristic of a DSS-friendly situation. Finally, if this sign-in is not the first of the day and the user signing-in is not the driver who signed-in last, then the new driver must confirm a driver changeover, and the previous driver must sign for his portion of the load and give a reason.

If this is the first sign-in of the day, then the user is sent to the Route Selection and then the Route Screen. The user may use the dropdown menu to select an existing route and forego manual data entry; if the user selects an existing route, then the route populates itself and may be edited. Both of these are examples of the system design satisfying Torres’ demand for an interface that minimizes demands on user input in a handheld user interface (HUI). To afford the HUI’s screen-size and stylus use, all editable text boxes require a mere tap to open a keyboard or numeric keypad; this includes entering a new route number in the Route Selection Screen.

Once a route is selected and accepted, the user goes to the Main Menu, where they can access the Route, Collect Load, Drop-Off Load, and Other Function Screens; additionally, he may receive messages about the temperature of the ice bath containing the milk samples and suggestions to initiate remote agitation of the next farm’s tank since the user is now within the agitation time of arriving at the next farm on the route. While early initiation of agitation may not affect user effectiveness, it is an ability internal to the system that helps the user

accomplish work and is, therefore, a functional affordance. Also displayed here and in every screen is the task bar, which is described in both the Usability Guidelines and Appendix B: Screen Descriptions.

The Collect Load Screen (CLS) appearance mock-up is in the diagram below. By far the most complicated individual component to the DSS, the CLS displays many of the design characteristics called for in previous chapters. Several desirable characteristics that have been pointed out on previous screens or future screens may not be pointed out here despite the fact that they are present.

Collect Load - Load Information			
	Tank		Temp
	<input type="text"/>	▼	<input type="text"/>
	Dipstick	Final Stick	Calc. Final Stick
	<input type="text"/>	<input type="text"/>	<input type="text"/>
	Pickup	Leave Behind	Truck Tot.
Wt.	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vol.	<input type="text"/>	<input type="text"/>	<input type="text"/>
	○ Partial Pickup		
	Sample	<input type="text"/>	
	Cancel		Accept

FIGURE 7 – Collect Load Screenshot

The Tank field auto-populates as “1” and increments if multiple tanks are pumped at one farm. Temp, Dipstick, and Sample, the asterisked fields, all require driver interaction; they would all appear raised and shaded like buttons, indicating they “want” to be “pushed”; making fields that require interaction raised and textured like a button afford the user’s senses. Now that the user associates these fields with buttons he knows that they afford tapping and that something will happen if they are tapped; that is a cognitive affordance. The user may tap the Temp field and then take the temperature reading, tap the dipstick reading and enter a numeric value in the keypad that pops up, and tap the sample field to generate a sample number and print those samples. (Extra labels are printed and extra samples are collected if the milk marketing agency managing this farm requests a special sample via the DSS.) The hauler may choose to make a partial pickup by selecting the Partial Pickup button, and then the gray fields above become visible. Final Stick is the stick reading after the pumlover. Leave Behind is the weight and volume corresponding to the Final Stick reading, and Pickup is the weight and volume equal to the original Dipstick weight and volume minus the Leave Behind weight and volume. If the Dipstick volume would overload the truck, the user is forced to make a partial pickup where the pickup will not overload the truck; if the user presses the button labeled “Calc Final Stick”, the system will automatically calculate the volume to be picked up and the level at which the user should leave the tank. If the Dipstick value would cause an overloaded truck the Truck Tot. weight and volume fields flash red until a valid value is entered in the Final Stick field or the Dipstick value is changed to an

acceptable reading. The DSS's ability to monitor proposed pickups and tank capacity displays Ginsberg and Stohr's DSS characteristic of extending the user's cognitive ability so he may be free to concentrate on other decisions and tasks. Values in all fields may be re-entered multiple times in case errors are made; this includes reprinting labels. Other features of this screen are its requirement that all data be filled-in and its ability to pump second tank at the same farm and track associate that milk; this level of traceability lends this system nicely to Butler, Herlihy, and Keenan's larger-scale DSS by allowing statistics like butterfat levels and milk volumes to be associated to individual sub-herds or even individual cows.

Another screen that bears attention is the Drop-Off Screen and its subscreens; while this series of screens contains many design features that display DSS characteristics, HUI-friendly principles, and different levels of affordances, its length and the redundancy of a thorough description of the techniques therein make a summary the best way to cover this section. The drop-off screen is used almost entirely by receivers at the processing plants, so the load and tank numbers are displayed in the title bar, since the receiver processes multiple tanks simultaneously and confusion must be minimized. All necessary approvals are afforded in this section, and if the receiver follows a non-recommended course of action, either by accepting a load that is not within the required parameters (e.g. the tank temp is one degree Fahrenheit over the maximum) or by rejecting a load that appears acceptable, then the DSS demands that he remark upon why he made that decision. This feature is used

several other times in this DSS and parallels one of the valuable features of the NICU DSS designed by Baxter et al..

Additional screens can be found in the Other Functions menu. One such screen is the Middle of the Day Sign-Out Screen; here a driver may approve his portion of the route and allow a new driver to complete the remainder of the route. Another screen found in Other Functions is the Disposal Screen, where the driver can dispose of milk at a location other than a standard processing plant (e.g. a dumpsite, a dog food manufacturer). The builders and programmers of the system may deem it necessary to put other screens in the Other Functions menu to manage the device and its peripherals (i.e. electronic thermometer, scanner) or screens that are only visible to qualified users.

RESULTS

In addition to improving operator effectiveness by removing opportunity for errors and violations, this system's design has capitalized on opportunities for increasing the efficiency of the operation, namely, allowing the operator to initiate agitation remotely at an optimal time. This virtually eliminates all wait time for the hauler (except for time in line at the processing center, which is another study); it also reduces an operator's temptation to violate agitation requirements, and by agitating completely the parties concerned with compensation for quality of milk are promised true results. To validate that remote agitation initiation's benefits outweigh its cost, an Arena simulation that mimics the hauler's new daily routine with and without remote agitation initiation has been developed. The simulation defines each farm, including distance and volume, travels to that farm, determines the agitation time required, and either does or does not begin agitation remotely. Then the hauler performs the pumprover process, the truck volume is incremented, and the truck either goes to the next farm or the plant (depending upon whether or not the tank has reached capacity). The following table details the variables involved and the expressions those variables represent; these variables have been developed through personal observation,

consultation of industry experts, and simulation experience on behalf of the designer. Several statistics recorded in the

TABLE II
SIMULATION VARIABLES

Variable	Expression
Farm Tank Volume	TRIA(300, 1000, 3000) gal.
Travel Time	TRIA(2, 30, 120) min.
Large Tank Agitation Time (>1250 gal.)	10 min.
Small Tank Agitation Time (< 1250 gal.)	5 min.
Pumpover Time	Volume / 100 gal. per min.
Setup/Clean-Up Time	TRIA(5,8,10)

course of the simulation confirm that simulation is a valid model of reality; these statistics include the length of the operator's day, final truck volume, average farm volume, and number of farms serviced. Keep in mind that this simulation represents the average route for the entire nation in which some trucks in California service one farm daily and some trucks in Kentucky service six farms daily. The simulation was run for 100 replications, and unique random number streams were used for each variable.

According to the simulation, implement early initiation of agitation reduces the average daily hauler wait time from .47 hours to .002 hours and overall hauler time on the job from 4.68 hours to 4.44 hours, a savings of 5% or 14.5 minutes per day. If the average hauling company's time (with respect to the hauler's basic routine) is worth \$1 per minute then that savings of \$14.50 and the

estimated daily cost of a cellular phone subscription (to initiate agitation) (phone equipment cost is assumed to be part of this value) of \$1 per day result in a yearly savings of \$4932.50 for the average hauling company.

CONCLUSIONS AND RECOMMENDATIONS

The goal of this study is to design a DSS, primarily the user interface and logic behind that interface, to support dairy collection and dairy security; this system should help ensure the security of the product, in conjunction with other systems protecting the truck tank itself, and the security and accuracy of the data. This document prescribes a robust, use-friendly system that satisfies those requirements and provides room for future increases in the sophistication of the dairy industry's data network. Various DSS characteristics and applications have been explored, and the best and most relevant characteristics discussed have been incorporated into the dairy collection DSS.

As the system is taken from this design through creation and implementation, the iterative process of human-centered design should continue to be followed; operators and other stakeholders should be encouraged to provide input; this input should be considered and incorporated into the design; and the user should be consulted again and again until the design is satisfactory. Once this system is implemented, great opportunities will exist for increased efficiencies on a massive logistical scale for the entire dairy industry, and those opportunities should be capitalized upon.

REFERENCES

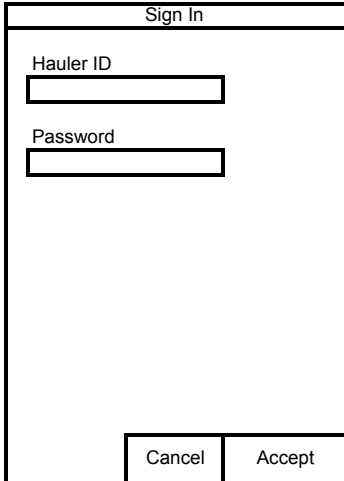
- Alter, S.L., "Decision Support Systems Current Practice and Continuing Challenges", Addison-Wesley Publishing Co., Reading, MA, 1980.
- Baxter, Gordon D., Monk, A.F., Tan, K., Dear, P.R.F., Newell, S.J., "Using cognitive task analysis to facilitate the organization of decision support systems into the neonatal intensive care unit", *Artificial Intelligence in Medicine*, 35, 2005.
- Bonczek, R.H., Holsapple, C.W., Whinston, A.B., "Foundations of Decision Support Systems", Academic Press, New York, 1981.
- Butler, Martin, Pat, H., Keenan, P.B., "Integrating information technology and operational research in the management of milk collection", *Journal of Food Engineering*, 70, 2005.
- Fiallos, Elman Lopez, "Decision Support Systems: Different Approaches", 1987.
- Geoffrion, A.M. and Powers, F., "Management Support Systems", *Decision Support Systems in a Data-Based, Model-Oriented, User-Developed Discipline*, Edited by W.C. House, PBI, New York, 1983.
- Hartson, H. Rex, "Cognitive, physical, sensory, and functional affordances in interaction design", *Behaviour and Information Technology*, 22, 5, September-October 2003.
- ISO 13407, International Organization for Standardization, Geneva, 1999.

- Keen, P.G.W.. “Interactive Computer Systems for Managers: A Modest Proposal”, Decision Support Systems, Edited by W.C. House, PBI, New York, 1983.
- Keen, P.G.W., Scott-Morton, M.S., “Decision Support Systems: An Organizational Perspective”, Addison-Wesley Series on Decision Support, Reading, MA, 1978.
- Keith, Angela Riccobono, “A Decision Support System for an Automobile Dealership”, 1985.
- Mitra, S.S., “Decision Support Systems: Tools and Techniques”, John Wiley and Sons, New York, 1986.
- Norman, Don., The Design of Everyday Things, Basic Books, New York, 1988.
- Spolsky, Joel, User Interface Design for Programmers, Apress, New York, 2001.
- Torres, R.J., Practitioner’s Handbook for User Interface Design and Development, Prentice Hall, New Jersey, 2002.

APPENDIX I

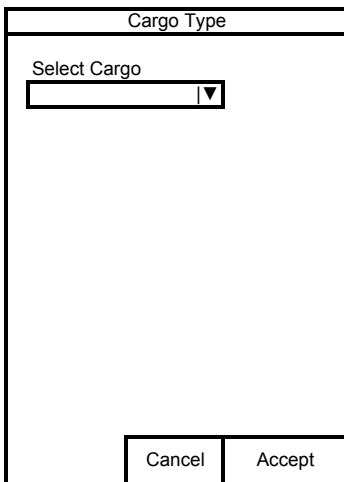
INTERFACE SCREENSHOTS

- ALL TAP-ABLE BOXES AND BUTTONS SHOULD BE RAISED AND GIVEN DIMENSION



The 'Sign In' screen features a title bar at the top with the text 'Sign In'. Below the title bar, there are two input fields: the first is labeled 'Hauler ID' and the second is labeled 'Password'. At the bottom of the screen, there are two buttons: 'Cancel' on the left and 'Accept' on the right.

- Auto-populate Hauler ID with last signed in user
- Hauler ID / Password combination must be valid.
If not valid, MESSAGE: "Enter valid ID and password."
If no valid, do not advance.
- Auto-associate hauler name and number, transport company name and number, milk marketing agency name and number



The 'Cargo Type' screen features a title bar at the top with the text 'Cargo Type'. Below the title bar, there is a dropdown menu labeled 'Select Cargo'. At the bottom of the screen, there are two buttons: 'Cancel' on the left and 'Accept' on the right.

- Default: Grade A Milk
- Only select at beginning of day
- Associate with today's pickups and drop-offs

Truck Information	
Scan Transport Tank	<input type="text"/>
Scan Tractor	<input type="text"/>
<div style="display: flex; justify-content: flex-end; gap: 10px;"> Cancel Accept </div>	

- Scan Tank textbox is flashing.
- Tap and scan transport tank
- If transport tank is not a straight truck prompt for tractor scan
MESSAGE: "This is not a straight truck. Tap and scan tractor."
- If transport tank is not scanned, do not allow advance
- If transport tank is not a straight truck and tractor is not scanned, do not allow advance.
- If advance is not yet allowed, "Accept" button should be grayed out.

NOTE: If transport tank has been the destination tank at a pumpover station then associate transferred milk tickets with this load and be aware of volumes as result of pumpover

Route Selection	
Select or Enter Route Number	<input type="text"/>
<div style="display: flex; justify-content: flex-end; gap: 10px;"> Cancel Accept </div>	

- Select route from dropdown or tap and enter new
- Default to driver's last route
- If no route selected, do not allow advance
MESSAGE: "Select Route Number."

LOCKED - Sign In	
Hauler ID	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="Cancel"/> <input type="button" value="Accept"/>	

- Auto-populate Hauler ID with last signed in user
- Hauler ID / If not valid, MESSAGE: "Enter valid ID and password."
If no valid, do not advance.
- If driver diff POPUP: "Are you sure you want to take over this route?"
If NO, return to "LOCKED" screen
If YES, prompt for original hauler signature.
Allow advance if signature given
If cancel, return to "LOCKED" screen

Main Menu
<input type="button" value="Route Menu"/>
<input type="button" value="Collect Load"/>
<input type="button" value="Drop-Off Load"/>
<input type="button" value="Other Functions"/>

- Other Func Scanner functions
Change Driver function
- If sample temperature exceeds 40° F
BLINKING MESSAGE: "Sample temperature over 40° F!"
- If sample temperature below 32° F
BLINKING MESSAGE: "Sample temperature under 32° F!"
- If within agitation time of next pick-up destination, GOTO "Agitation" screen

Messages' Area				
Buttons' Area				
<table border="1" style="width: 100%;"> <tr> <td style="width: 25%;">1</td> <td style="width: 25%;">2</td> <td style="width: 25%;">4</td> <td style="width: 25%;">3</td> </tr> </table>	1	2	4	3
1	2	4	3	

- 1 Power Off - Save all confirmed data and shut-off (must be signed in)
- 2 Lock - Lock device (must be signed in)
- 3 Battery Lev Green/Yellow/Red as level descends (split in thirds)
Tap for MESSAGE: "Battery Level: XX%" for 5 seconds
- 4 Transport 1 Green/Yellow/Red as level ascends (split in thirds)
Tap for MESSAGE: "Tank Level: XX%" for 5 seconds

ROUTE MENU

Route	
Prod. No.	Producer Name
111111	AAAAAAA
222222	BBBBBBB
333333	CCCCCCC
444444	DDDDDDD
555555	EEEEEEE
9999999	Processor Name

Insert
Remove
Move

Cancel
Accept

Already-visited farms are grayed out and may not be modified.

Processor, like farm no.s and names, is a modifiable field. Distinguish with different color.

If not selected, default to no # and name "Select Processor"

INSERT

- Tap "Insert"
- MESSAGE: "Tap destination to insert new farm in front of."
- INSERT remains depressed and make Prod. No.s **bold**
- User taps a farm or processor name
- "Insert Farm" screen appears

MMA's may wish to control. If so, MMA must be able to modify this and lock other users out of that ability.

Haulers may list pumpover-type locations as pick-up destinations

- MESSAGE: "Tap Farm Number to add farm."
- Farm Number textbox's outline flashing until tapped
- Farm Name not modifiable by hauler
- Auto-populate name associated with entered number
- If data complete and hauler taps Accept insert this farm before the farm selected on Route Screen. Exit to Route Menu.
- "Accept" only valid if valid data entered; otherwise, grayed out
- If hauler taps Cancel, Exit to Route Menu

Route Menu - Insert Farm

Farm Number

Farm Name

Cancel
Accept

DIRECT EDITING

- Hauler taps Farm Number on Route List
- POPUP: "Are you sure you want to replace Farm #####?"
- NO --> Return to Route Menu
- YES --> Screen like Insert Farm screen appears but already auto-pop.'ed
- Same rules apply, except farm is replaced not pushed back

Do you want to remove farm 111111 AAAAAA?	
NO	YES

REMOVE

- Tap "Remove"
- MESSAGE: "Select destination to remove."
- User taps a farm (no processor) and confirmation popup appears
- NO --> exit to Route Menu
- YES --> Remove farm from route and exit to Route Menu.

Route Menu - Processor Selection	
Processor Number	<input type="text"/>
Processor Name	<input type="text"/>
<input type="button" value="Cancel"/> <input type="button" value="Accept"/>	

Tapping Processor Number

- "Route Menu - Processor Selection" screen appears.
- Auto-populate if a processor is associated with route.
- MESSAGE: "Tap and enter processor number"
- Processor Number textbox outline flashing until tapped
- "Not yet known" is option when text box is tapped
- Name not modifiable by hauler
- "Accept" only valid if valid data entered; otherwise, grayed out

MOVE

- Tap Move button
- MESSAGE: "Tap destination to move"
- Farm Numbers in bold.
- Hauler taps farm number (not processor)
- MESSAGE "Tap destination to place 111111 before."
- Processor number and all Farm Numbers but 111111 bold.
- Hauler taps farm number or processor number
- POPUP: "Place 111111 AAAAAAA in front of 444444 DDDDDDD?"
- NO --> change nothing; return to Route Menu
- YES --> make change; return to Route Menu

ACCEPT

- Tap Accept button
- Not clickable unless at least one farm and processor chosen
- If tapped and either is missing, MESSAGE: "One farm and one processor required."
- If tapped and data complete, POPUP: "Save these changes?"
- NO --> Return to Route Menu
- YES --> Save changes and exit to Main Menu

CANCEL

- Tap Cancel button
- POPUP: "Discard these changes?"
- NO --> Return to Route Menu
- YES --> Exit to Main Menu

COLLECT LOAD

Collect Load - Scan BTU

Scan Bulk Tank Unit

Cancel

Accept

- Text Box outline flashing until tapped
- MESSAGE: "Tap textbox and scan BTU."
- Hauler taps textbox and scans BTU
- CANCEL --> Exit to Main Menu
- ACCEPT, only valid if valid BTU scanned in, grayed-out otherwise
- ACCEPT --> Advance to Appearance Check

Collect Load - Appearance

Accept?

- YES
- NO

Reason for Rejection

REASON LIST

Reject

Cancel

Accept

- Default to Yes
- If NO tapped, Reason list appears
- If YES tapped afterward, Reason list disappears
- If NO tapped, Confirm is not clickable until reason selected.
- CONFIRM --> Load Information screen.
- If clicked upon when invalid, MESSAGE: "Select Reason for Rejection."
- REJECT, operable if NO selected and a reason is chosen
 POPUP, "Reject this load based on appearance?"
 YES, Note reject and consider this part of route complete
 NO, do nothing

Collect Load - Load Information		
Tank	Temp	
<input type="text"/>	<input type="text"/>	
Dipstick	Final Stick	Calc. Final Stick
<input type="text"/>	<input type="text"/>	<input type="text"/>
Pickup	Leave Behind	Truck Tot.
<input type="text"/>	<input type="text"/>	<input type="text"/>
Wt.	Vol.	
<input type="text"/>	<input type="text"/>	
<input type="radio"/> Partial Pickup		
Sample	<input type="text"/>	
<input type="button" value="Cancel"/>		<input type="button" value="Accept"/>

* Fields hauler normally modifies

Temperature Out of Range
Temperature is __°F. What action will you take?
<input type="button" value="Accept"/> <input type="button" value="Reject"/> <input type="button" value="Remeasure"/>

- Accept: Accept in spite of temp and note
- Reject: Require confirmation. Record reject and reason
- Remeasure: Return to Load Info Screen and clear temp text box

Reject this load based on temperature?	
<input type="button" value="Do NOT reject."/>	<input type="button" value="Reject"/>

- TANK defaults to one; increment automatically is another tank is collected
- TEMP is required; tap and scan; if unacceptable, TEMP POPUP.
- If TEMP is already entered, may tap and remeasure
- Dipstick is required; tap and enter reading
- Pickup weight and volume are calculated and populated
- Truck Tot. Wt. and Vol. are current + proposed pickup
- Sample is required; hauler taps and a number is generated and labels are printed
- If labels already printed, POPUP: "Print new labels?" (If YES, discard old number and print new labels)
- If milk marketing agency entered request for special sample(s) the night before, then POPUP reads "Collect extra sample."
- Extra label of same # is printed
- When dipstick reading is entered, if proposed pickup is greater than remaining transport tank capacity, display totals in red and MESSAGE: "Adding this load exceeds truck's capacity."
- *1 - Final stick and Leave Behind become visible and Partial Pickup is checked
 - "Calc. Final Stick" button appears until tapped
 - Final stick is auto-populated with dipstick reading that equals the amount to be left behind if transport tank is filled completely. Leave Behind and Truck Totals are calculated based upon this.
 - Final stick is modifiable. Leave Behind and Truck Totals are recalculated
 - Final Stick cannot be changed to a value less than that which meets the volume restraints of the transport tank.
 - MESSAGE: "You must leave more behind."
 - Hauler may re-enter original Dipstick reading at any time.
 - If Pickup exceeds capacity, begin at *1 and Final Stick is blank
 - If Pickup does not exceed none of the gray buttons become invisible and Partial Pickup is unselected.
- If Partial Pickup is selected, gray boxes become valid (except Calc. Final Stick Button). Recalc Leave Behind and Truck Totals accordingly when Final Stick readings are entered.
- If deselected, gray boxes invisible
- Cannot deselect if proposed pickup exceeds remaining transport tank capacity
- Once per month, when temp is measured, POPUP:
 - "Does tank thermometer equal measured temperature?"
 - Record answer.
- Accept is only valid when asterisked fields are filled and partial load data, when applicable, is acceptable

CANCEL

- Tap Cancel
- POPUP: "Cancel this pickup?"
- YES --> exit to Main Menu but do not modify Route Menu
- NO --> return to Load Information screen

ACCEPT

- Tap Accept
- POPUP: "Complete this load?"
- NO --> return to Load Information screen
- YES --> Save Load information THEN GOTO Additional Load Popup

Additional Load

Is there another load to pick up
at this location?

YES, pick up another load here.

NO

Additional Load

- YES --> Enter a fresh Load Information screen and increment Tank by 1
- NO --> Exit to Main Menu

DROP-OFF

Drop-Off Location

Location No.

1

2

Destination Truck

3

Cancel Accept

Non- Pumpover

- Location Number textbox outline is flashing until tapped
- MESSAGE: "Tap Location No. and scan location barcode"
- 1: auto-populate with associated location name
- 2: auto-populate with associated location type
- 3: if location type is "Pumpover Station", the "Destination Truck"

- Accept only operable with valid location number scanned in
- Cancel --> Main Menu
- Accept --> Drop-Off - Receiver screen

Pumpover

- Accept only operable with valid location and destination truck numbers scanned-in
- Cancel --> Main Menu
- Accept --> Drop-Off - Pumpover screen

Drop-Off - Receiver

Receiver No.

3

Location No.

1

2

Cancel Accept

- Receiver Number textbox outline is flashing until tapped
- MESSAGE: "Tap Receiver No. and scan receiver barcode"
- 3: auto-populate with associated receiver name
- Accept only operable if valid receiver number scanned
- Cancel --> Back to Drop-Off Location screen
- Accept --> Drop-Off Checklist

- Display Location information from previous screen
(May NOT edit here.)

Drop-Off Checklist [Load and Truck No.]	
1. Meets Location Requirements-----	<input type="checkbox"/>
2. Samples Received-----	<input type="checkbox"/>
3. Milk Ticket Approval-----	<input type="checkbox"/>
4. Truck Tank Approval-----	<input type="checkbox"/>
5. Load Approval-----	<input type="checkbox"/>
6. Sample Approved (double-tap to "X")-----	<input type="checkbox"/>
7. Disposition of Milk-----	<input type="checkbox"/>
8. Wash Truck-----	<input type="checkbox"/>
9. Sign-Off-----	<input type="checkbox"/>
Reject	Cancel
Accept	

THIS PORTION PERFORMED BY RECEIVER

- Meets Location Requirements is checked by computer if all pickup locations' GPS coordinates matched those on record
- Check boxes should be raised like buttons; tap those to get to menus related to check items
- CANCEL, POPUP: "Are you sure you want to discard this info.?"
YES, discard and return to Drop-Off - Receiver screen
NO, return to Drop-Off Checklist
- ACCEPT
 - If all boxes checked, Save and Export all data
Display "END OF DAY" for 20 seconds and shut-down
 - If not all boxes checked, POPUP: "Not all boxes are checked. Are you sure you want to accept this load."
YES, Save and Export all data
Display "END OF DAY" for 20 seconds and shut-down
NO, return to Drop-Off Checklist
- REJECT
 - If not all boxes checked, POPUP: "Are you sure you want to REJECT this load?"
YES, Save and Export all data, noting Rejection and allow Comment
NO, return to Drop-Off Checklist
 - If all boxes are checked, POPUP: "This load is acceptable. Are you sure you want REJECT it?"
YES, Save and Export all data, noting Rejection and require Comment
NO, return to Drop-Off Checklist

Samples Received

Samples Received			
Farm	Tank	Sample No.	
321321321	1	#####	<input type="checkbox"/>
321321156	1	#####	<input type="checkbox"/>
123232122	1	#####	<input type="checkbox"/>
123232122	2	#####	<input type="checkbox"/>
Reject	Cancel	Accept	

- CANCEL, save no information and exit to Drop-Off Checklist screen
- REJECT, only operable if at least one box is unchecked
places red "X" in box 2 on Drop-Off Checklist
- ACCEPT, only operable if all boxes checked
places black check in box 2 on Drop-Off Checklist

Milk Ticket Approval

Milk Ticket	
Hauler # _____	Name _____
Company # _____	Name _____
Tank ID _____	Arrival Time _____
Start Time _____	Finish Time _____
Agency # _____	Name _____
Lic./Permit # _____	
Total Load Wt. _____	
Producer # _____	Name _____
Tank _____	Wt. _____ Temp. _____
Time _____	Dipstick _____
BTU _____	Driver Sign. ? _____
Sample # _____	
<div style="border: 1px solid black; padding: 2px; display: inline-block;">Initials</div>	
Reject	Cancel
Accept	

- CANCEL, revert to Drop-Off Checklist
- ACCEPT, return to Drop-Off Checklist and place black check in box 3
- REJECT, return to Drop-Off Checklist and place red "X" in box 3

A

One section like this for every tank of every producer picked up on this truck load. This section must be scrollable and go from A to B.

B

Milk Tank Approval

Milk Tank Approval	
1. Locks in good condition?	<input type="checkbox"/>
2. Tank acceptable?	<input type="checkbox"/>
3. Wash ticket matches this:	<input type="checkbox"/>
<div style="border: 1px solid black; padding: 2px; display: inline-block;">LAST WASH TICKET NUMBER</div>	
Reject	Cancel
Accept	

- REJECT, only valid if at least one box unchecked
- REJECT --> return to Drop-Off Checklist and place red "X" in box 4
- ACCEPT, only valid if all boxes checked
- ACCEPT --> return to Drop-Off Checklist and place black check in box 4
- CANCEL --> return to Drop-Off Checklist

Load Approval

Load Approval					
1. Appearance	No <input type="checkbox"/>	Yes <input type="checkbox"/>			
2. Temperature	<input type="text"/>				
<table border="1"> <tr> <td>Reject</td> <td>Cancel</td> <td>Accept</td> </tr> </table>			Reject	Cancel	Accept
Reject	Cancel	Accept			

- CANCEL --> return to Drop-Off Checklist
- Temperature, tap and scan
- REJECT and ACCEPT only valid if response given for items 1 and 2
- REJECT, POPUP: "Reject based on this?"
 - YES, return to Drop-Off Checklist and place red X in box 5
 - NO, do nothing
- ACCEPT, only valid if item 1 is YES and item 2 is between 32 and 40 degrees Fahrenheit
- If temperature is outside acceptable range, MESSAGE: "Temperature is outside safe range."

SAMPLE APPROVED

- Receiver checks if test results are acceptable.
- Double-tap to reject

DISPOSITION OF MILK

- Refers to disposition within plant
- Select from drop-down list in POPUP that also has ACCEPT and CANCEL buttons

WASH TRUCK

Wash Truck				
1. Truck Washed	No <input type="checkbox"/>	Yes <input type="checkbox"/>		
2. Method	<input type="text"/>			
3. Brand	<input type="text"/>			
4. Wash Tag Number	<input type="text"/>			
<table border="1"> <tr> <td>Cancel</td> <td>Accept</td> </tr> </table>			Cancel	Accept
Cancel	Accept			

- CANCEL, return to Drop-Off Checklist
- ACCEPT, operable if a response is given for item 1.
- If item 1 is YES then a response must be given for item 2, 3, and 4.
 - MESSAGE: "What method was used?"
- ACCEPT, if item 1 is YES--> return to Drop-Off Checklist and place black check in box 8
- ACCEPT, if item 1 is NO--> return to Drop-Off Checklist and place red "X" in box 8

Wash Tag Number

- Tap and scan

*Only accessible if all other boxes have a response SIGN OFF

SIGN OFF				
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 2px;">HAULER</th> </tr> <tr> <td style="height: 20px;"></td> </tr> <tr> <td style="height: 20px;"></td> </tr> </table>	HAULER			
HAULER				
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 2px;">RECEIVER</th> </tr> <tr> <td style="height: 20px;"></td> </tr> <tr> <td style="height: 20px;"></td> </tr> </table>	RECEIVER			
RECEIVER				
<table border="1" style="display: inline-table; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">Cancel</td> <td style="padding: 2px 10px;">Accept</td> </tr> </table>	Cancel	Accept		
Cancel	Accept			

- ACCEPT, only valid if both signature received
- CANCEL--> return to Drop-Off Checklist
- ACCEPT--> return to Drop-Off Checklist, place black check in box 9

PUMPOVER				
Pumpover Checklist [Load and Truck No.]				
1. Meets Location Requirements-----	<input type="checkbox"/>			
2. Samples Received-----	<input type="checkbox"/>			
3. Load Approval-----	<input type="checkbox"/>			
4. Sample Approved-----	<input type="checkbox"/>			
5. Pumpover Volume-----	<input type="checkbox"/>			
6. Wash Truck-----	<input type="checkbox"/>			
7. Sign-Off-----	<input type="checkbox"/>			
<table border="1" style="display: inline-table; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">Reject</td> <td style="padding: 2px 10px;">Cancel</td> <td style="padding: 2px 10px;">Accept</td> </tr> </table>	Reject	Cancel	Accept	
Reject	Cancel	Accept		

- 1 as normal
- 2 as normal
- 3 as normal
- 4 go to Pumpover Sample Screen
- 5 go to Pumpover Volume Screen
- 5 as normal
- 6 as normal
- Cancel as normal
- Accept as normal
- upon accepting, transfer milk ticket information and testing checklist information to destination truck tank, including volume

Required: 2, 3, 4 (if partial pumpovers done or likely), 5, 7

NOTE: Milk Marketing Agency may flag that anything at this pumpover location or individual trucks must test samples (#5)

NOTE: Destination truck will have to perform no actions to receive pumpover. If pumpover occurs while dest, truck is mid-route They must have pumpover as current pick-up destination, GPS location must be logged along with scanning location barcode

Pick-Up Pumpover	
Pumpover Location	
<input type="text"/>	
Cancel	Accept

Pumpover Sample		
Tester	<input type="text"/>	
Test Method	<input type="text" value="IV"/>	
Test Result	<input type="checkbox"/> Pass <input type="checkbox"/> Fail	
Reject	Cancel	Accept

- CANCEL --> return to Pumpover Checklist
- REJECT only valid if data complete and result is "Fail"
- ACCEPT only valid if data complete and result is "Pass"
- REJECT requires confirmation POPUP

Pumpover Volume

Pumpover volume

Cancel
Accept

- CANCEL --> return to Pumpover Checklist
- Pumpover Volume, tap and enter
- IF pumpover volume + destination truck volume exceeds destination truck capacity
 POPUP: "This will exceed destination truck's capacity. Would you like to pumpover the maximum allowable amount?"
 YES--> autopopulate pumpover volume with
 Dest. Truck capacity - current total volume
 NO--> return to Pumpover checklist

On log-in, if truck has not been washed since last end-of-day or within require time period (72 hours) operator may not advance beyond this screen.

Cleaning Required

Location No.

Name

Receiver

Name

1. Method ▼

2. Brand ▼

3. Wash Tag Number

Shutdown
Accept

- Tap Location Number and scan
- Location Name auto-populates based on scan
- Tap Receiver Number and scan
- Receiver Name auto-populates based on scan
- Tap and scan wash tag number
- All data must be filled-out for ACCEPT to be operable

Other Functions
New Driver
Load Disposal
Main Menu

New Driver
Original Driver <input type="text"/>
Reason for leaving <input type="text" value="▼"/>
Signature <input type="text"/>
<input type="button" value="Cancel"/> <input type="button" value="Sign Out"/>

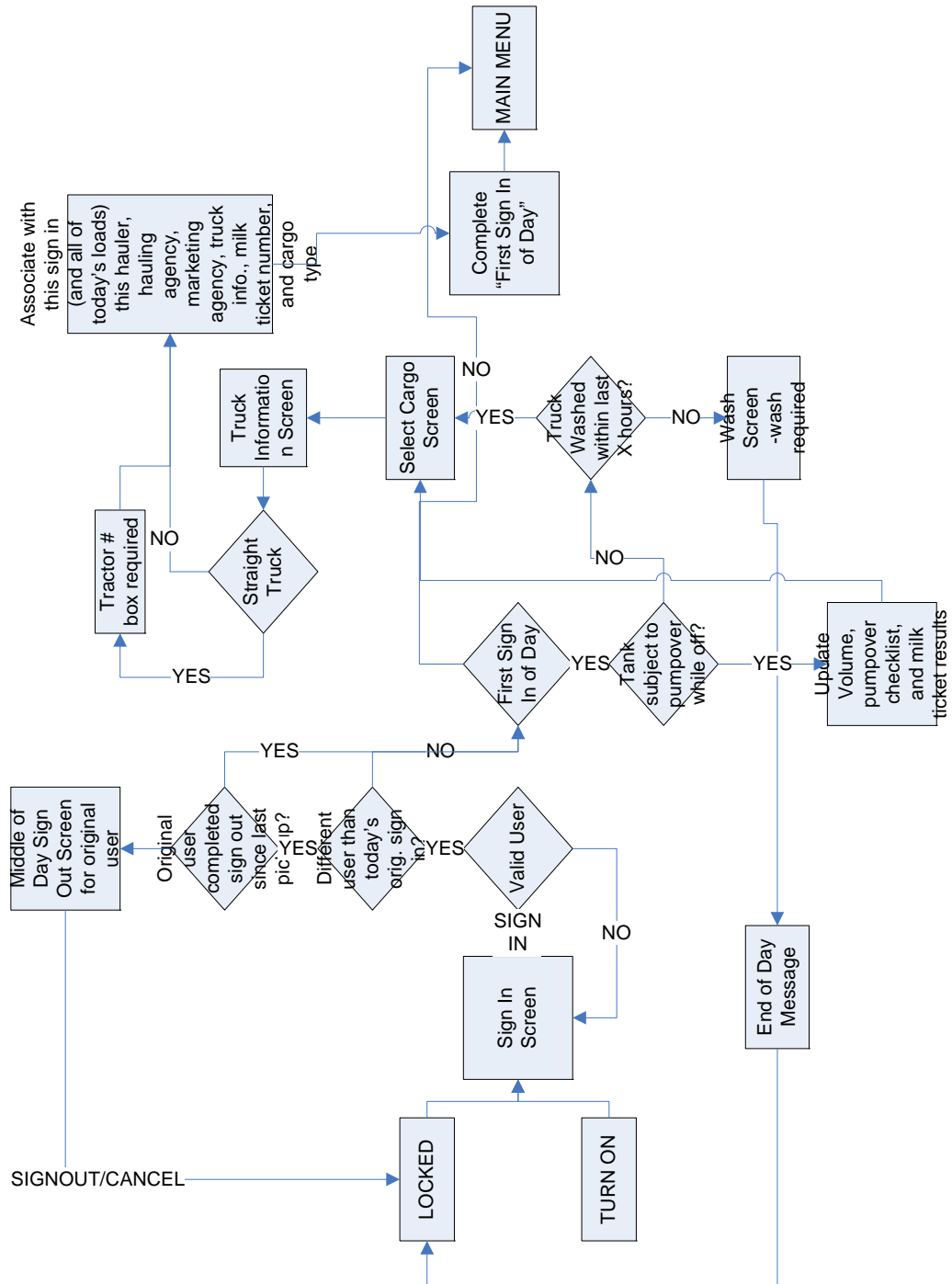
- CANCEL--> Return to Other Functions Menu
 - ACCEPT, only valid if all three pieces of data present
 - ACCEPT--> Locked Screen
- NOTE: same hauler may resume task

Load Disposal	
Disposal Type	<input type="text"/>
Location	<input type="text"/>
Signature	<input type="text"/>
Cancel	Sign Out

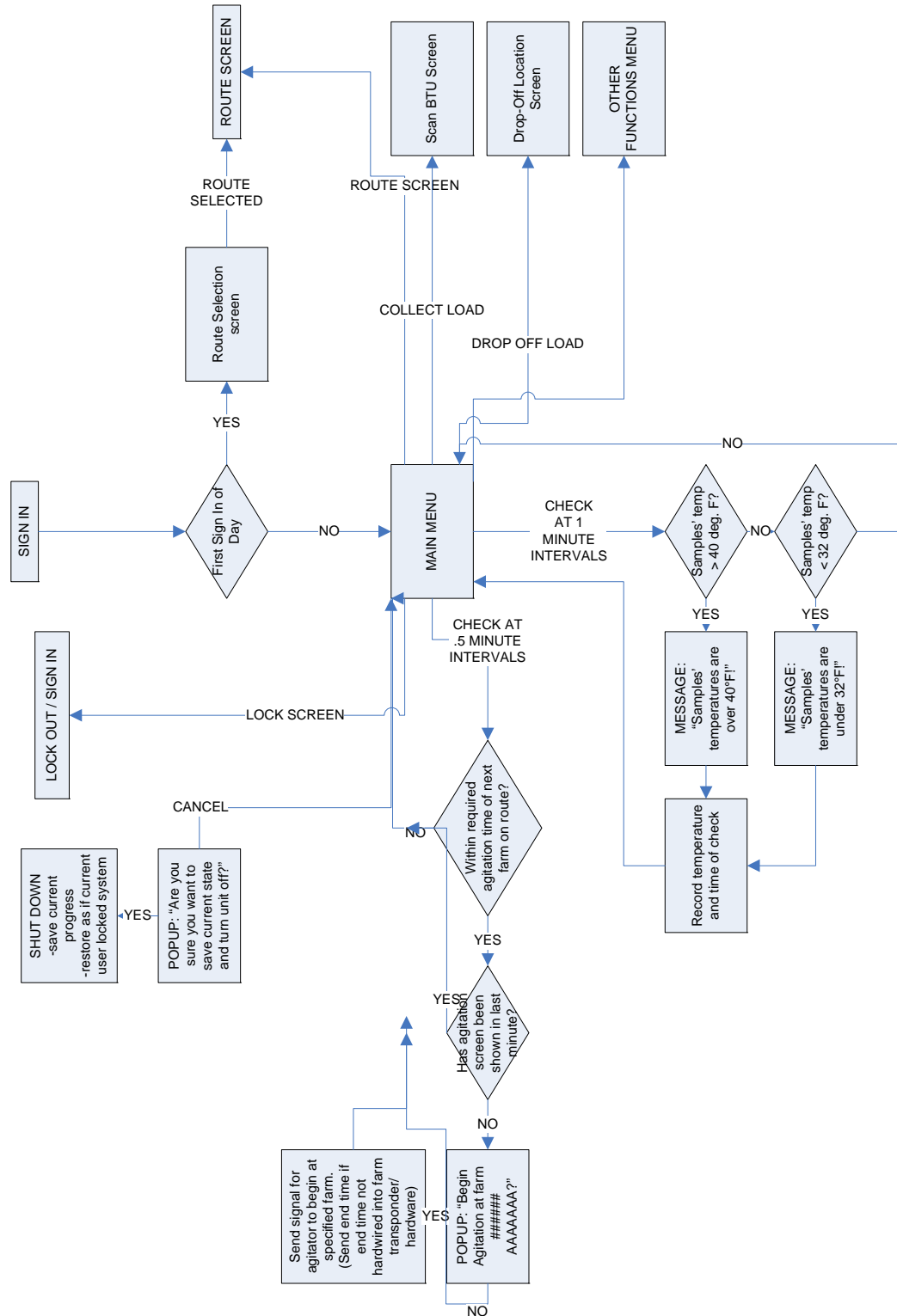
- May select or tap and edit disposal type
- May select or tap and edit location
- Record time internally
- ACCEPT, only operable if all three pieces of data present
- CANCEL--> return to Other Functions Menus
- ACCEPT--> POPUP: "Wash truck or signout?"
 - Wash Truck --> GOTO Cleaning Required screen
 - Sign-out --> EOD procedure

APPENDIX II

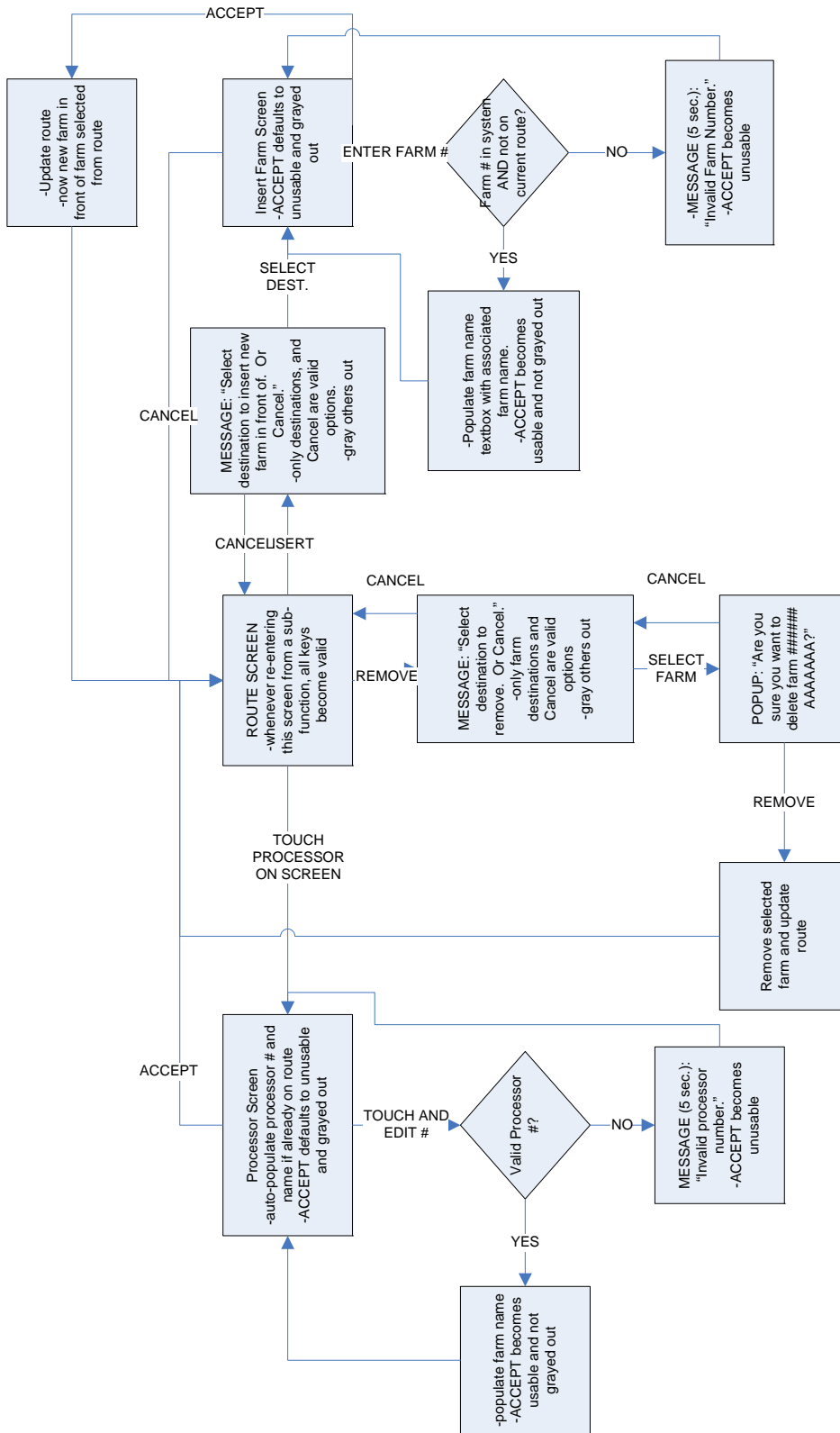
SIGN IN



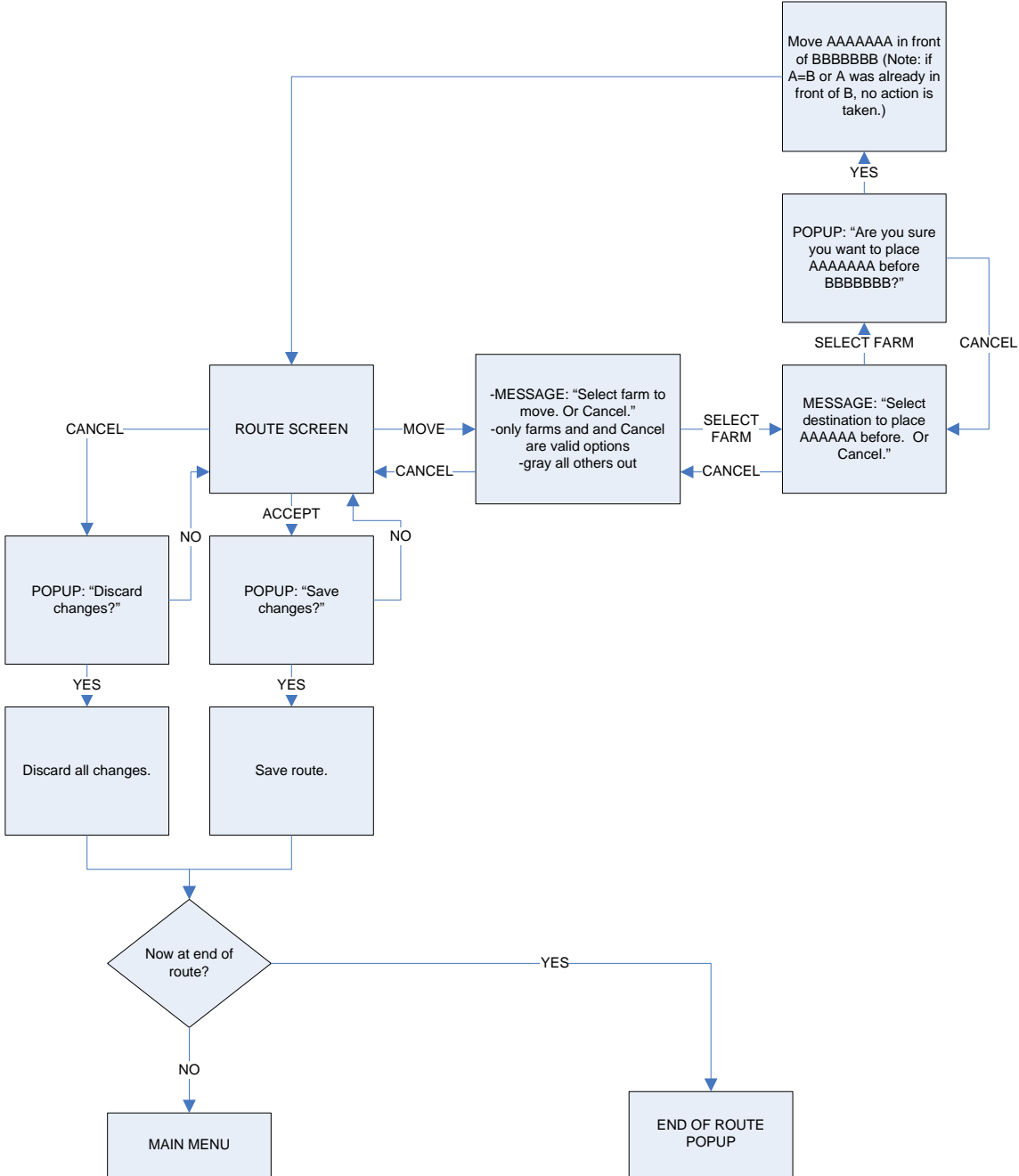
MAIN MENU



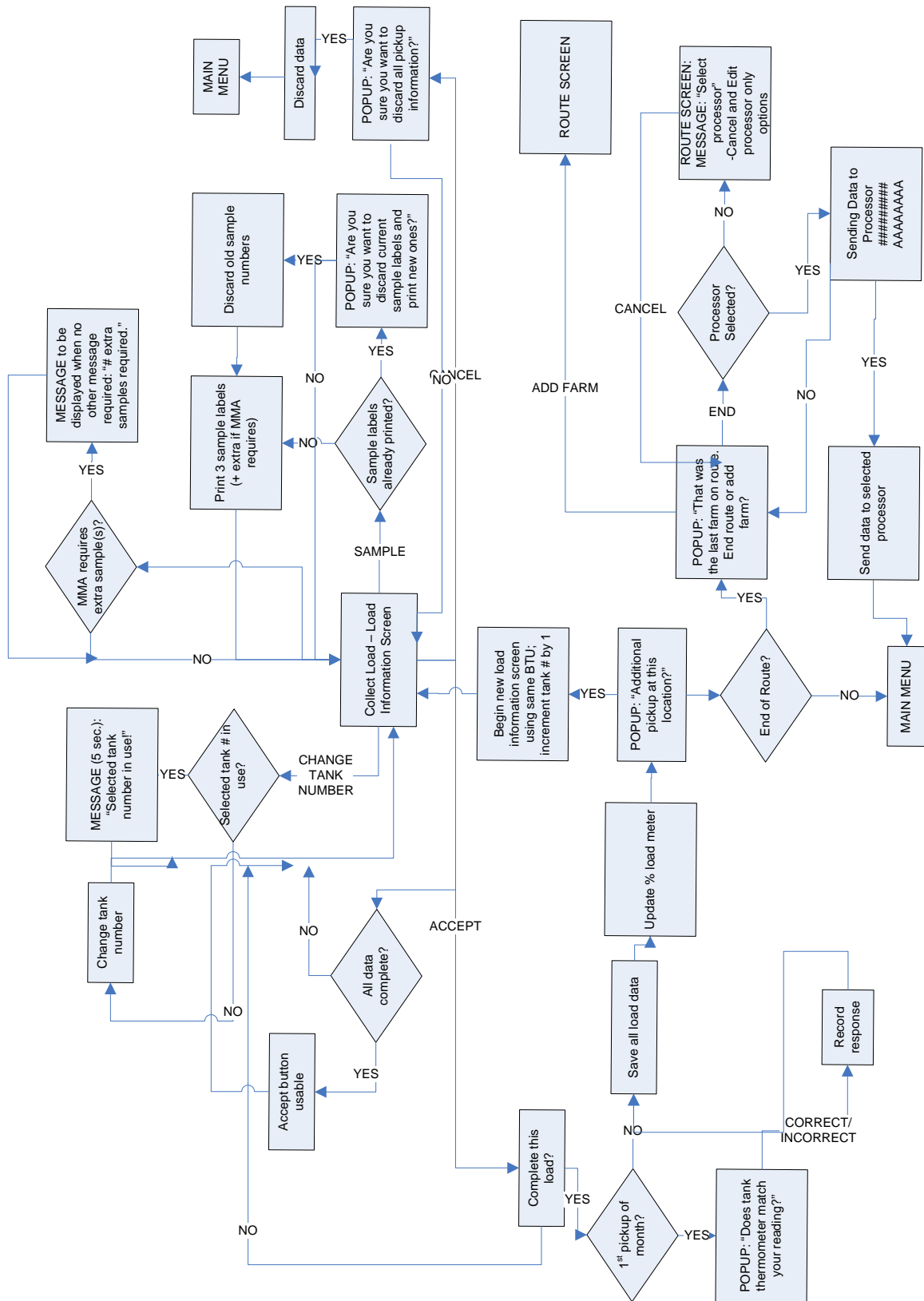
ROUTE SCREEN - 1



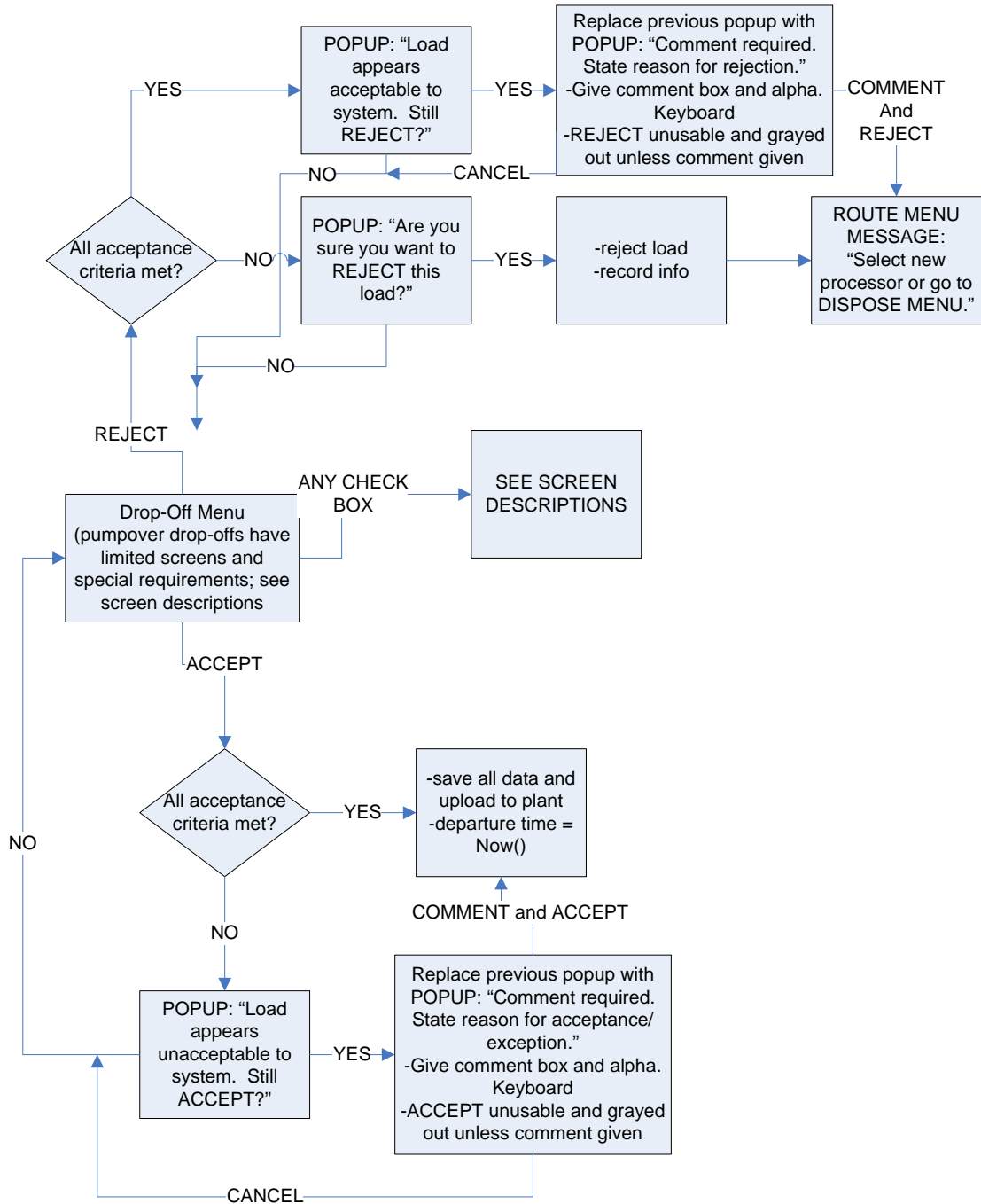
ROUTE SCREEN – 2



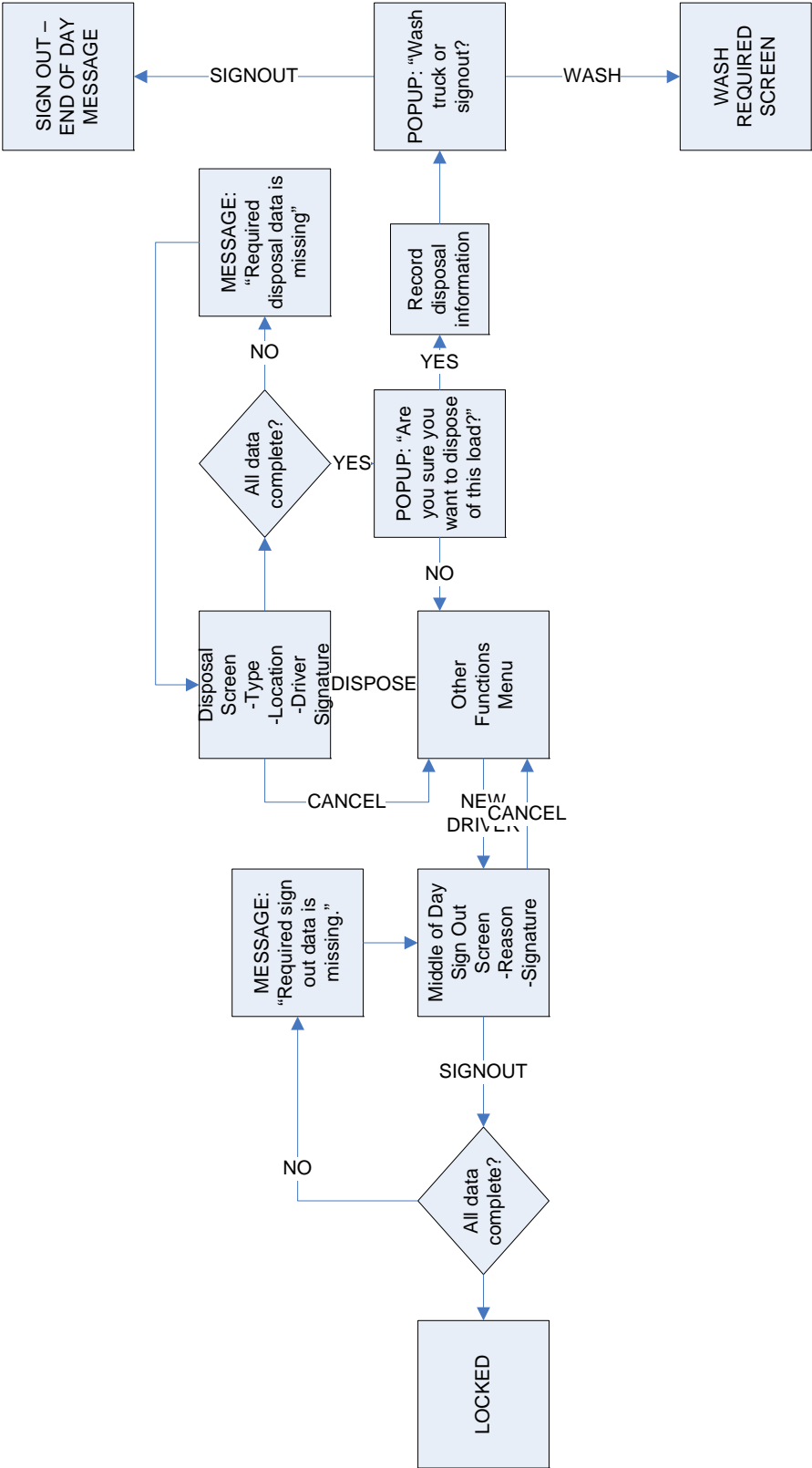
COLLECT LOAD - 2



DROP-OFF MENU



OTHER FUNCTIONS



CURRICULUM VITAE

NAME: Charles Milan Braidich

ADDRESS: 4643 Autumn Leaves Drive
Columbus, IN 47203

DOB: Newark, Ohio – June 30, 1983

EDUCATION
& TRAINING: B.S., Industrial Engineering
University of Louisville
2001-2005

M.Eng., Industrial Engineering
University of Louisville
2005-2006

AWARDS:

PROFESSIONAL SOCIETIES:

Institute of Industrial Engineers

PUBLICATIONS:

NATIONAL MEETING PRESENTATIONS

REFERRED JOURNALS

BOOKS AND SYMPOSIA

INVITED PRESENTATIONS