ICE TRACKER Project Requirements

A CS532 team is tasked to develop an aut[illegible] sy[illegible] [illegible]n Company. The automated system, referred to as *ICE TRACK*, [illegible] initial tracking capability by the end of thesemester. The clients for this system have indicated [illegible] *ICE TRACK* should support the following business functional areas:

a. Order Entry

b. Inventory Management

c. Shipment Tracking

d. Trouble Ticket Management

The following sections identify the general system-level requirements and the high-level requirements for each of the functional areas.

General Requirements

Graphical User Interface (GUI): *ICE TRACK* shall utilize a full graphical user interface that provides multiple, moveable, and sizeable windows. When a user completes all required steps in a transaction and commits that transaction to the database, *ICE TRACK* shall respond within 5 seconds.

Operational Environment: *ICE TRACK* shall be designed to support multiple users in networked Windows environment. The initial version may be a single user version, but the database application must be able to migrate to multi-user/shared environment accessible over a local area network.

Security Access: Protecting the security of information maintained by *ICE TRACK* shall depend primarily on using object-based security. A password shall be required to sign on, and shall be validated by the system before the user can gain access to it.

Roles and Privileges: Once validation has taken place, information stored in the system shall determine which objects a user shall be able to access, by way of pre-assigned roles and privileges, and what actions he or she shall be able to take on those objects. *ICE TRACK* shall label all transactions with the user's name and the date and time.

Auto Timeout Logoff: To limit the opportunity for unauthorized persons to use signed-on but unattended, workstations, a signed-on user shall be required to sign off when he or she takes a break, leaves his or her workstation for an extended period of time, or ends his or her shift. *ICE TRACK* shall automatically log off any signed-on user who does not make an input into the system for a user-defined interval (e.g., five minutes).

Single Entry of Information: All information maintained by *ICE TRACK* shall be entered only once. When information has been entered, it shall be available for use in subsequent *ICE TRACK* transactions. (For example, once customer information has been entered, it is available to all other relevant *ICE TRACK* functions: order entry, shipment tracking, and trouble ticket management.)

Validation of Inputs: Information entered into *ICE TRACK* shall be validated on entry, and identified errors shall be resolved before additional information can be entered. The system edits necessary to perform this validation shall be controlled by a combination of user-defined and user-maintained parameters. A list of possible values for each field of entry will be provided to the operator by pressing the appropriate key sequence. These values will be user-maintained.

Entry and Processing of Names: *ICE TRACK* shall accept and correctly process hyphenated names in any name component, first, middle, and/or last. *ICE TRACK* shall also allow name searches using the standard wild card characters.

Modular Design: The system shall be modular in design. This is to permit addition of devices to the system, or the incorporation of new types of devices into the system, as needed. It shall also be possible to add, upgrade, or replace software modules in the system with relative ease.

Data Structure: It is to be expected that a customer may be associated with many orders and that an ice cream product will be associated with multiple orders. It is also expected that an order will contain multiple entries and may be related to multiple shipments. The system shall be designed, therefore, with both one-to-many and many-to-many structures, which is consistent with the use of a relational database to store information.

Display Printing: It shall also be possible to print any display presented to a user on an *ICE TRACK* workstation. Such display printouts shall include all header information, and shall be readable and usable by anyone familiar with *ICE TRACK* display and report conventions.

Reporting Requirements: All printed reports should have the capability of being viewed on the workstation before printing. Reports should be accessible as a whole or in parts.

On-Line Help Function: To assist personnel in learning to use *ICE TRACK*, and to help them solve problems encountered while using the system, *ICE TRACK* shall incorporate an on-line help function. To facilitate access by the *ICE TRACK* end-user, the on-line help function shall be context sensitive. It shall also be sufficiently detailed to provide answers to most of the questions asked by *ICE TRACK* end-users.

The help facility will include overall system functions, transaction descriptions, screen descriptions, and data fields. Users will have the ability to update the help screens under security control provided by the system.

ICE TRACK shall allow a user to ask for and obtain help for each entry or inquiry module or screen, while the user is in that module or screen. The "help" system shall provide essential information for the module or screens purpose and usage.

ICE TRACK shall allow the user to ask for and obtain "help" for each individual field, while the cursor is in that field, for each entry and inquiry screen. The "help" function shall give the user a brief explanation of the field requirement(s) for that particular field, including whether the field is mandatory (i.e., data MUST be entered in the particular field, if applicable).

Screens/User Interface: The system will be menu driven. There will be standardized movement from screen to screen without having to return to the main menu. System messages will be standardized.

Ability to Access Any Screen: *ICE TRACK* shall allow the user the ability to access any screen with the minimal use of sub-menuing.

Prototype Design / Process: All customization, modifications, or new development will be accomplished using JAD (Joint Application Design) methodology. Prototypes will be utilized to ensure desired functionality.

Control of Information Entry Sequence: The system shall control the presentation of data collection screens and/or menus, and the collection of information. In normal cases, information entry should occur in a predetermined sequence. For typical cases, a user shall have the ability to select any desired screen to enter information out of the normal sequence.

Error Message: If a user attempts to input information out of sequence, or the information entered is inconsistent with requirements for information entry, the entry shall be rejected, and an error message shall indicate: 1.) the reason for the rejection, and 2.) the minimum requirements that must be met for the entry to be accepted.

Processing Block Until Error Corrected: As an example of the requirement that information be input in the proper sequence, failure to complete the customer entry shall result in almost all other new order entry functions failing to work because there is no name record on which to attach information.

Control Entry Modification: The user shall be able to return to a screen already processed to add optional information, or to modify information previously entered. As a minimum, return to a screen already processed shall be accomplished by selecting the desired screen from a menu.

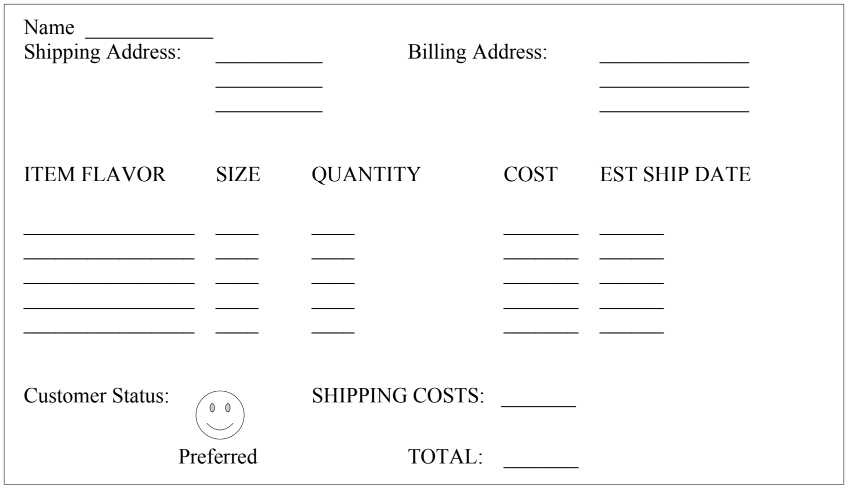
Modification Consistency Checks: *ICE TRACK*, nevertheless, shall not allow the modification of previously entered information that creates inconsistent conditions. For example, a user cannot change a shipping address after the product has been shipped.

User Cancellation of Action Before Completion: *ICE TRACK* shall also allow the user to cancel an action at any time prior to his or her completion of that action.

Restoration of Previous Data: Upon determining that the user has canceled an action, *ICE TRACK* shall restore any interim information entries the user has made to their status prior to the initiation of the action.

Order Entry Requirements

- The order entry subsystem shall automate the order entry process to the greatest extent possible.
- The order entry subsystem shall maintain customer information that includes customer name, shipping address, billing address, and customer status (e.g., preferred customer, ok customer, shaky customer—doesn't always pay on time).
- The order form should allow multiple entries (line items) for ice cream orders by ice cream flavor, size of container, and number of cases. Ice cream flavor and container size should be administrator-maintained lists of the available flavors and sizes. The order form should also indicate the customer name, the delivery address, the billing address, type of shipping desired, the desired receipt date, the expected ship date, the cost for each line item (ice cream order), the shipping costs, and the total cost of the order (total of the line item costs plus shipping costs). The clients have provided a preliminary mock-up of the order entry form they would like. They would like this evaluated and revised if necessary.



Name ____________
Shipping Address: __________ Billing Address: ______________
__________ ______________
__________ ______________

ITEM FLAVOR	SIZE	QUANTITY	COST	EST SHIP DATE
________________	____	____	_______	______
________________	____	____	_______	______
________________	____	____	_______	______
________________	____	____	_______	______
________________	____	____	_______	______

Customer Status: ☺ Preferred

SHIPPING COSTS: _______

TOTAL: _______

- The order entry process shall check the current inventory to determine the availability of the item. If the available quantity is not sufficient to fulfill the order, the system shall display a message indicating the amount of product currently available and the expected date that additional inventory will be available. For available inventory, the order entry process shall reserve the desired quantity. The order entry subsystem shall allow a user to generate a notification to the trouble ticket management subsystem if he/she is unable to take a customer order due to insufficient inventory.
- If the customer is a preferred customer, there should be no limit on the order size. If the customer is an ok customer, the system should limit the order total cost to $3,000. If the customer is shaky, the system should limit the order total cost $500.
- The system should allow the user to print out a copy of the order that can be sent to the customer as an invoice.
- If the customer cancels or modifies an order after it has been shipped (or partially shipped), the customer shall be billed for any shipping charges incurred.
- The system shall allow the user to enter the date that payment(s) were received.
- The system shall allow the user to print out a report showing all outstanding invoices, the amount due on each invoice, the date payment is due, and the number of days (if any) the payment is overdue. The report shall be sorted by customer name.

Inventory Management

- ❑ The inventory management subsystem shall track the ice cream inventory for *Tom and Adam's* Ice Cream Company. The inventory management subsystem shall track product items by flavor and unit/package size. The system shall maintain a user-definable list of unit/package sizes supported by the *Tom and Adam's* Ice Cream Company.
- ❑ The inventory management subsystem shall maintain a list of available inventory as well as inventory that is present but committed to unfilled orders.
- ❑ The inventory management subsystem shall maintain a schedule for the receipt of planned inventory products.
- ❑ The inventory management subsystem shall allow the user to directly add products to the inventory as well as change the status from "planned" to "actual" for those items in the schedule of planned inventory products. When inventory is added to the system, the quantity and type values shall be updated for the appropriate product items.
- ❑ The inventory management subsystem shall allow the user to delete product items from the inventory.
- ❑ The inventory management subsystem shall maintain a log that identifies the user associated with each inventory add or delete transaction.
- ❑ The inventory management subsystem shall maintain information about the disposition of inventory (e.g., why an item was moved out of the current inventory). The initial list of dispositions shall include the following:

  a. SHIPPED and the associated order number

  b. DEFECTIVE PRODUCT

  c. PRODUCT SPOILAGE
- ❑ The inventory management subsystem shall send a notification to the trouble ticket management subsystem for all products removed from inventory due to defects or spoilage.
- ❑ The inventory management subsystem shall allow the user to generate statistical summary reports showing the disposition of the ice cream inventory for a specified week, month, or year.

Shipment Tracking

- ❑ The shipment tracking system shall track the status of all *Tom and Adam's* Ice Cream shipments.
- ❑ The shipment tracking subsystem shall include the date the order was shipped, number of shipment boxes associated with the order, whether a full or partial order was shipped, the expected delivery date, the actual date the order was delivered, and the method by which each shipment box was shipped. The methods for shipment shall be a user-maintained list of shipment options possible for the shipment destination (geographic region).
- ❑ If the customer cancels or modifies an order after it has been shipped (or partially shipped), the customer shall be billed for any shipping charges incurred.
- ❑ The shipment tracking subsystem shall maintain a list of authorized shipping vendors by type of shipping and geographic region. The list of shipping vendors shall be user-maintained and include shipping rate information and vendor rating (preferred, OK, poor).
- ❑ The shipment tracking subsystem shall allow the user to query the status of a shipment or set of shipments by customer name, destination (specific address or city), shipment date, or expected delivery date.
- ❑ The shipment tracking system shall maintain information on lost or damaged products. The information maintained on lost or damaged products shall include the identification of the shipment box(es), the problem with the shipment, the shipping vendor, and the cost associated with the lost or damaged shipment. The shipment tracking system shall send a notification to the Trouble Ticket Management subsystem for each lost or damaged shipment.
- ❑ The shipment tracking subsystem shall maintain information on reshipments due to lost or damaged products.
- ❑ The shipment tracking subsystem shall maintain shipping status information on all shipments for at least 30 days after full delivery and receipt of payment, whichever is later.
- ❑ The shipment tracking subsystem shall allow the user to print out a shipment status report that shows
  - a. All active orders ready for shipment.
  - b. All active orders in the system that have been shipped but not yet delivered.
  - c. Summary of lost or damaged shipments during user-specified period of time.

Trouble Ticket Management

- ❑ The trouble ticket management subsystem shall maintain information on both internal production/order fulfillment/shipping problems and customer-facing problems.
- ❑ The trouble ticket management subsystem shall maintain the following information on reported internal production/order fulfillment/shipping problems:
  - a. Date problem reported
  - b. Source: Name of user reporting problem or name of user associated with process reporting the problem
  - c. Date problem detected
  - d. Date problem resolved
  - e. Type of problem
  - f. Description of problem
  - g. Problem status (open, being worked, closed)
  - h. Problem resolution
- ❑ The trouble ticket management subsystem shall maintain the following information on reported customer-facing problems:
  - a. Date problem reported
  - b. Source: Name of customer reporting problem
  - c. Date problem detected
  - d. Date problem resolved
  - e. Type of problem
  - f. Description of problem
  - g. Problem status (open, being worked, closed)
  - h. Problem resolution
- ❑ The trouble ticket management subsystem shall maintain a user-maintained list of typical problems reported either internally or by customers. The trouble ticket management subsystem shall also maintain a user-maintained list of actions to take for each typical problem in order to facilitate problem resolution.
- ❑ The trouble ticket management subsystem shall accept problem reports from the order entry, inventory management, and shipment tracking subsystems and automatically insert them into the problem tracking database.
- ❑ The trouble ticket management subsystem shall allow the user to enter trouble ticket information into the system.
- ❑ The trouble ticket management subsystem shall allow the user to query the trouble ticket database by customer name, type of problem, problem status, date problem reported, date problem resolved.
- ❑ The trouble ticket management subsystem shall allow the user to request the following reports:
  - a. Summary of all open trouble tickets by problem type, date reported, or date closed.
  - b. Statistical summary report of number of problems reported, average time to close, average number of open problems/day, and average number of problems being worked/day for a user-specified period of time.
- ❑ The trouble ticket management subsystem shall allow the user to export trouble ticket information in a format that can be imported to an Excel spreadsheet for further analysis and reporting.