

The Krusty Krab Inventory Management System
Analysis Phase Deliverables:
System Proposal

Alyssa Bronge, James George, Chandler Stegen

Dr. Kaefer

INFS 347

October 25, 2022

Table of Contents

I. Executive Summary	3
II. System Request	4
III. Work Plan	5
IV. Feasibility Analysis	6
V. Requirements Definition	8
VI. Use Cases	9
VII. Process Models	14
VIII. Data Model	16
IX. Appendices	17

Executive Summary

Here we have presented a finalized system proposal as the analysis phase deliverable for the Krusty Krab Inventory Management System. We have included an updated system request which includes the updated scope of the project. We cut down on some features of the system such as forecasting and communication with suppliers as that would have made the project too big. We updated the work plan based on our completion of the planning phase. As well as an updated feasibility analysis taking into consideration the updated scope of the project. For parts of the analysis phase, we have included the requirements definition which was used to help determine our use cases, process models, and data model. We have conducted six use cases based on those requirements which will be beneficial to building the system successfully. Additionally, we have a context and level 0 data flow diagram based on those use cases and an entity relationship diagram explaining how our data will be stored for the system. Finally, we have attached additional resources in the appendix to determine the requirements for the new system, including an employee questionnaire.

System Request – Inventory Management Project

Project sponsor: Eugene Harold Krabs – CEO/Owner

Business Need: The project addresses the need to be able to more efficiently manage inventory within the restaurant. This system will account for waste/spoilage/theft that has occurred. Allow for servers to simultaneously create checks and update inventory through the point of sale (POS) system. The system will also help streamline the reorder process by making it a computerized activity.

Business Requirements: Servers will be able to input orders into the POS system and the system will update the current inventory based on that information. A manager in the back will have access to the system to account for any miscellaneous dilemmas that require an update in the inventory. Different users will have different access to the system to prevent any potential theft or mismanagement. All ingredients for recipes of menu items need to be determined to effectively manage inventory through the POS system. The system needs to send a notification to the manager when the inventory level of a particular ingredient is low and should be reordered.

Business Value: Creates a single system that allows for easily accessible live inventory information. Will mitigate the labor costs of physically tracking inventory. Provides a more efficient process for determining inventory levels by allowing servers to both create tabs and update inventory in one singular process within the same system. The system will help eliminate stockouts and optimize our reorder process. Having a stronger grasp of live inventory can help when making on-the-spot decisions regarding serving certain menu items and the efficiency of food preparation within the restaurant.

Conservative estimates of tangible values to the business include

- \$200,000 in reduced inventory costs
- \$130,000 in reduced labor costs for inventory management
- \$100,000 in revenue from increased customer sales

Special Issues or Constraints: Due to the nature of the restaurant business we must account for the freshness of food and expiration dates when implementing the system so we will utilize the first-in-first-out (FIFO) system when deciding which inventory to use. The System needs to be able to manually update through the backend if a known mistake was made by a server, for example when using POS. Managing and tracking certain foods may prove difficult.

Work Plan

Task ID	Task Name	Assigned To	Duration (days)	Estimated		Actual		Duration Variance	Dependency	Status
				Start Date	End Date	Start Date	End Date			
1	Planning Phase		13	9/6/2022	9/27/2022					Completed
1.1	System Request	Chandler	2	9/6/2022	9/8/2022					Completed
1.2	Feasibility Study	Chandler	5	9/9/2022	9/16/2022					Completed
1.2.1	Technical Feasibility	Chandler	2	9/9/2022	9/12/2022					Completed
1.2.2	Economic Feasibility	Chandler	2	9/13/2022	9/15/2022					Completed
1.2.3	Organizational Feasibility	Chandler	1	9/16/2022	9/16/2022					Completed
1.3	Project Plan	Alyssa	6	9/19/2022	9/27/2022					Completed
1.3.1	Work Plan	Alyssa	2	9/19/2022	9/21/2022					Completed
1.3.2	Staffing Plan	Alyssa	1	9/22/2022	9/22/2022					Completed
1.3.3	Standards List	James	1	9/23/2022	9/23/2022					Completed
1.3.4	Risk Assessment	James	2	9/26/2022	9/27/2022					Completed
Task ID	Task Name	Assigned To	Duration (days)	Start Date	End Date	Start Date	End Date	Duration Variance	Dependency	Status
2	Analysis Phase		16	10/3/2022	10/24/2022					Open
2.1	System Proposal	Chandler								Open
2.1.1	Requirements Definition	Chandler	4	10/3/2022	10/6/2022					Open
2.1.1.1	Functional Requirements	Chandler	2	10/3/2022	10/4/2022					Open
2.1.1.2	Non-Functional Requirements	Chandler	2	10/5/2022	10/6/2022					Open
2.1.2	Use Cases	Alyssa	3	10/7/2022	10/9/2022					Open
2.1.3	Process Model	Alyssa	4	10/12/2022	10/15/2022					Open
2.1.4	Data Model	James	3	10/16/2022	10/20/2022					Open
2.2	Appendices	James	2	10/21/2022	10/24/2022					Open
Task ID	Task Name	Assigned To	Duration (days)	Start Date	End Date	Start Date	End Date	Duration Variance	Dependency	Status
3	Design Phase		31	1/6/2023	2/17/2023					Open
3.1	Develop database design document	Alyssa	9	1/6/2023	16-Jan					Open
3.1.1	Staging database design	Alyssa	9	1/6/2023	1/16/2023					Open
3.1.2	Suspense database design	Alyssa	9	1/6/2023	1/16/2023					Open
3.2	Develop reject-handling design document	Alyssa	9	1/17/2023	1/29/2023				1.1.1, 1.1.2	Open
3.2.1	Reject-handling engine design	Alyssa	9	1/17/2023	1/29/2023					Open
3.3	Develop OLAP design document	James	9	1/17/2023	1/29/2023				1.1.1, 1.1.2	Open
3.3.1	Universe design	James	9	1/17/2023	1/29/2023					Open
3.4	Develop OLAP design pt. 1	Chandler	8	1/10/2023	1/21/2023					Open
3.4.1	High-priority reports design	Chandler	8	1/10/2023	1/21/2023					Open
3.5	Develop application design document	Alyssa	9	1/17/2023	1/29/2023					Open
3.5.1	Group consolidation + corporate reporting maintenance application design	Alyssa	9	1/17/2023	1/29/2023					Open
3.6	Extract, transform, load (ETL) design document	James	2	1/30/2023	1/31/2023				1.5	Open
3.6.1	Data export utility design	James	2	1/30/2023	1/31/2023					Open
3.7	Application design document	Chandler	27	1/16/2023	2/11/2023					Open
3.7.1	Web entry application UI design	Chandler	26	1/16/2023	2/10/2023					Open
3.7.2	Web entry application UI design sign-off	Chandler	1	2/11/2023	2/11/2023					Open
3.7.3	Web entry forms and database model validation	Alyssa	11	1/15/2023	1/29/2023					Open
3.8	Functional requirements document	James	9	1/20/2023	1/30/2023					Open
3.8.1	Application design	James	9	1/20/2023	1/30/2023					Open
3.8.1.1	User authentication	James	4	1/20/2023	1/23/2023					Open
3.8.1.2	Call logging	Chandler	2	1/24/2023	1/27/2023					Open
3.8.1.3	Search	Chandler	3	1/28/2023	1/30/2023					Open

Inventory Management System Feasibility Analysis

Patrick Star has created the following feasibility analysis for The Krusty Krabs' inventory management project. The highlights of the feasibility analysis are as follows:

Technical Feasibility

The inventory management system is technically feasible, although it is associated with some risks.

Krusty Krabs' risk regarding familiarity with the operation of the inventory management system is moderate.

- Servers are already familiar with operating the Point-of-Sale (POS) system and the new system will not require any additional electronic processes on their end.
- Management already has familiarity with the POS system as they operate the business aspect of the system and should be easily able to understand accessing and assessing the new system's information and tasks.

Krusty Krabs' risk regarding familiarity with the technology of the inventory management system is moderate.

- Our POS system was internally created so our engineers and developers have experience building a system of this caliber.
- Combining an inventory management system into the already existing POS system presents some risks. We do not want to worsen any of the already existing functionalities or make those more difficult to operate.

The Project size is considered relatively high.

- We are looking to completely rehaul our inventory management by computerizing/automating it into our POS system.
- Will require up to five employees to plan, oversee, and create the system.
- Will need to create a recipe costing structure to be implemented into the system.

The compatibility with Krusty Krabs' existing internal infrastructure should be good:

- The system will be based on the already existing technical infrastructure used in order processing and employee tracking.
- There is already a solid internet infrastructure within the restaurant that supports current systems.
- By hosting our system in-house, we do not have to rely on uncertainties regarding third-party services.

(Economic and Organizational Feasibility can be found on the next page)

Inventory Management System Feasibility Analysis (*continued.*)**Economic Feasibility**

A cost-benefit analysis was performed; see the attached spreadsheets in the Appendix for details. Conservative estimates show that the Inventory Management Project has a solid likelihood of contributing to The Krusty Krabs' bottom line.

ROI over 5 years: 49%

NPV over 5 years: \$63,282

Break-even occurs after 2.45 years

Intangible Costs and Benefits

Increased customer service as more optimized inventory management will reduce stockouts of items allowing for menu items to be more available when serving customers. Increased operational transparency through live inventory tracking will enable management to feel more comfortable addressing inventory situations when a decision needs to be made. We will have a higher quality menu as we can optimize our inventory making up better equipped to serve food that is fresher and therefore tastier for our customers.

Organizational Feasibility

From an organizational perspective, the project has high risk.

- Top Management Support: Top executives of the company have great support for the project
- Project Champion: Harold Eugene Krabs is a well-regarded and astute business executive who has been eager to implement a better inventory management system
- Organization Management: Unequivocally, managers support the creation of the new inventory management system. This will allow them to better understand the living environment of the restaurant. As well as allowing them to make more informed decisions when it comes to managing inventory. Some risks are associated with management being able to effectively use the system.
- Chefs: High risk associated with chefs having to comply with recipe guidelines for live inventory tracking to be effective. As well as accounting for any spoilage/waste that occurs within the food preparation process.
- Servers: High risk associated with servers accurately inputting order information into the system. The system must not force them to spend additional time associated with manually tracking inventory.

Functional Requirements (Manager):

1. Track Food
 - 1.1. Identify menu recipes
 - 1.2. Insert ingredients into the system
 - 1.3. Be able to easily add new ingredients to the system
 - 1.4. Insert menu items into the system
 - 1.5. List ingredient requirements associated with menu items
2. Reorder Inventory Request
 - 2.1. Management will decide and insert in the system what qualifies as low inventory
 - 2.2. System will identify when product inventory is low
 - 2.3. System will send a notification to the manager that product inventory is low
3. Manager Inventory Access
 - 3.1. Manager logs into the system
 - 3.2. Manager selects they would like to personally edit inventory
 - 3.3. Systems ask the reason for the update (theft, waste, spoilage)
 - 3.4. Manager selects inventory item
 - 3.5. Manager selects the quantity of an item
 - 3.6. System updates inventory levels
4. System Inventory Updates
 - 4.1. Management orders inventory
 - 4.2. System tracks order
 - 4.3. When the order arrives, the system updates the inventory

Functional Requirements (Server):

5. Processing Customer Orders
 - 5.1. Server will input orders into the POS system
 - 5.2. System will update the inventory based on the information entered

Non-Functional Requirements:

1. Operational
 - 1.1. The system should be on tablets and one main desktop to be used by management
 - 1.2. The system should connect to printers wirelessly
2. Performance
 - 2.1. The system should provide response times of 5 seconds or less
 - 2.2. The system should be updated with new inventory levels every 3 mins
3. Security
 - 3.1. All managers should be assigned a username and have a unique password to login to the access system
 - 3.2. Passwords must be 16 characters with at least one capital letter, at least one number, and a special character (?, !, @, *,)
4. Cultural and Political
 - 4.1. Company policy says that all computer equipment is purchased from Dell

Use Cases

Use Case Name: Manager Lists Ingredients	ID: 01	Priority: Medium
---	---------------	-------------------------

Brief Description: Manager lists ingredients being used by restaurant for menu items			
Actor: Manager			
Trigger: Manager needs to update ingredients			
Type External Temporal			
Preconditions:			
Normal Course		Information for Steps	
1.0 Update Ingredients list			
1. Manager needs to update the ingredients list		Ingredient	
2. Manager enters the system and chooses to update ingredients			
3. System asks the manager how they want to update the ingredients			
4. Manager selects between two update options (add/delete) (see Alternative Course 1.1)		Ingredients List	
7. System updates datastore ingredients			
Alternative Course(s):			
1.1 Manager selects between update options			
1a. Manager selects to add a new ingredient			
2b. System displays text box to insert new ingredient name			
3c. System confirms new ingredient has been added			
1b. Manager selects delete an ingredient			
2b. System displays a text box for the manager to look up the ingredient			
3b. Manager selects the ingredient to be deleted			
4b. System confirms ingredient has been deleted			
Postconditions:			
Exceptions:			
Summary Inputs	Source	Summary Outputs	Destination
Ingredient	Manager	Ingredients List	Ingredients datastore

Use Case Name: Menu Item Tracking	ID: 02	Priority: High
--	---------------	-----------------------

Brief Description: All menu items with ingredients must be entered into system and updated when there are new menu items added/deleted/modified	
Actor: Manager	
Trigger: Need to update menu items	
Type <input type="checkbox"/> External <input type="checkbox"/> Temporal	
Preconditions:	
Normal Course 1.0 Manager updates menu items <ol style="list-style-type: none"> 1. Manager selects update menu items in system 2. System prompts manager with three options to update menu items (add/modify/delete) (See Alternative Course 1.1) 3. Manager selects update menu 4. System asks manager to confirm menu item modification 5. Manager confirms modification 6. System updates menu items in inventory data store 	Information for Steps Menu item Ingredient Ingredient quantities Menu Item Menu Item ingredient quantities
Alternative Course(s): 1.1 Manager selects between menu item update options <ol style="list-style-type: none"> 1a Manager selects to add menu item option 2a. System displays text to add new menu item information 3a For each menu item added <ol style="list-style-type: none"> Manager enters the name of the menu item Manager enters ingredients required for menu item Manager enters quantity of each ingredient menu item 2b. Manager selects modify menu item option 3b. System displays text box for manager to enter menu item to modify 4b. For each menu item modification <ol style="list-style-type: none"> Manager has option to modify menu item name Manager has the option to modify menu item ingredients Manager has the option to modify menu item ingredient quantity 2c. Manager selects delete menu item option 3c. System displays text box to enter which menu item to delete 4c. Manager selects menu item to be deleted 	
Postconditions: <i>(Use case continued next page)</i>	
Exceptions: When the manager enters an ingredient for a menu item, they must assign an ingredient quantity to that menu item	

Summary Inputs	Source	Summary Outputs	Destination
Menu Item	Manager	Menu Item	Inventory datastore
Ingredient	Manager	Menu Item ingredient quantities	Inventory datastore
Ingredient quantities	Manager		

Use Case Name: Manager Updates Inventory Manually		ID: 03	Priority: High
Brief Description: The shift manager must be able to access the system to manual update the inventory			
Actor: Manager			
Trigger: Food spoilage/waste/theft			
Type External Temporal			
Preconditions:			
Normal Course		Information for Steps	
1.0 Manager updates food loss			
1. Manager logs into the system			
2. Manager selects what item they would like to personally edit the inventory		Inventory Information	
3. Systems ask the reason for the update (theft, waste, spoilage)			
4. Manager selects inventory item		Inventory List	
5. Manager selects the quantity of ingredient			
6. Systems update inventory levels		Updated Inventory	
Alternative Course(s):			
Postconditions:			
Exceptions:			
Summary Inputs	Source	Summary Outputs	Destination
Inventory information	Inventory	Updated Inventory	Inventory
Inventory list	Inventory		

Use Case Name: Server Inventory Update	ID: 04	Priority: High
Brief Description: The server inputs orders into the system to in order for the inventory database to update the amount of inventory the restaurant has		

Actor: Server			
Trigger: Server has an order that the inventory needs to be able to update			
Type External Temporal			
Preconditions:			
Normal Course 1. Server will input orders into POS system 2. System will update inventory based on information entered		Information for Steps Order ID Inventory Information Updated Inventory	
Alternative Course(s):			
Postconditions:			
Exceptions:			
Summary			
Inputs	Source	Outputs	Destination
Customer Order	Server	Updated Inventory	Inventory
Inventory Information	Inventory		

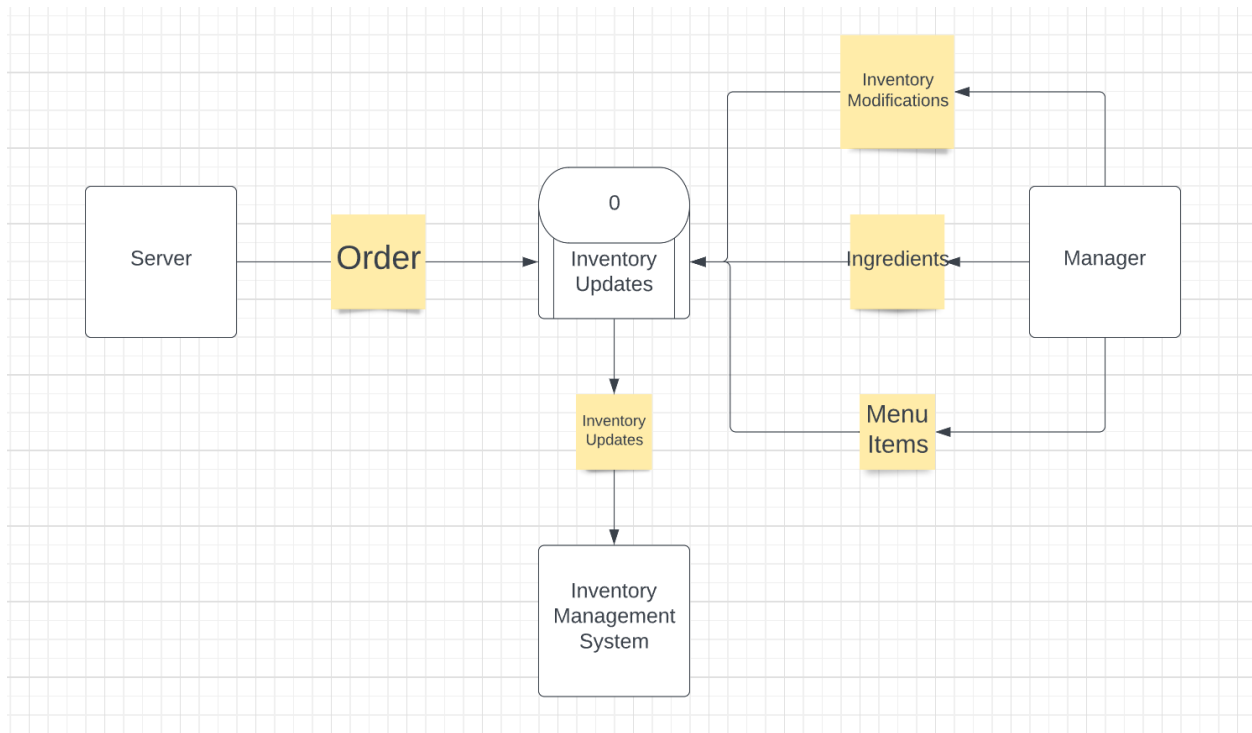
Use Case Name: Inventory Update Request		ID: 05	Priority: Low
Brief Description:			
This use case describes the receipt of low inventory levels sent to managers			
Actor: Manager			
Trigger: Inventory levels are low			
Type External Temporal			
Preconditions:			
Normal Course		Information for Steps	
1.0 Record Receipt of Low Inventory			
1. Manager retrieves low inventory notification			
2. System records date/time notification is received		Low Inventory level notification	
3. Manager records the quantity necessary to order.		Quantity needed for order	
Alternative Course(s):			
Postconditions:			
Exceptions:			
Summary Inputs	Source	Summary Outputs	Destination
Low inventory level notification	Inventory update request	Quantity needed for order	Manager

Use Case Name: Management Orders Inventory		ID: 06	Priority: High
Brief Description: This use case describes the order of new inventory placed by the manager.			
Actor: Manager			
Trigger: Quantity needed for order			
Type " External " Temporal			
Preconditions:			

Normal Course	Information for Steps		
1.0 Finalize Order Quantity	Quantity needed for order		
1. Manager opens the quantity needed for order doc.			
2. Manager inserts ordered quantity in system			
3. System records the completion date/time in the inventory data storage			
4. Inventory is stored in database	Ordered quantity update		
Alternative Course(s):			
Postconditions:			
Exceptions:			
Summary Inputs	Source	Summary Outputs	Destination
Quantity needed for order	Inventory update request	Ordered quantity update	Inventory

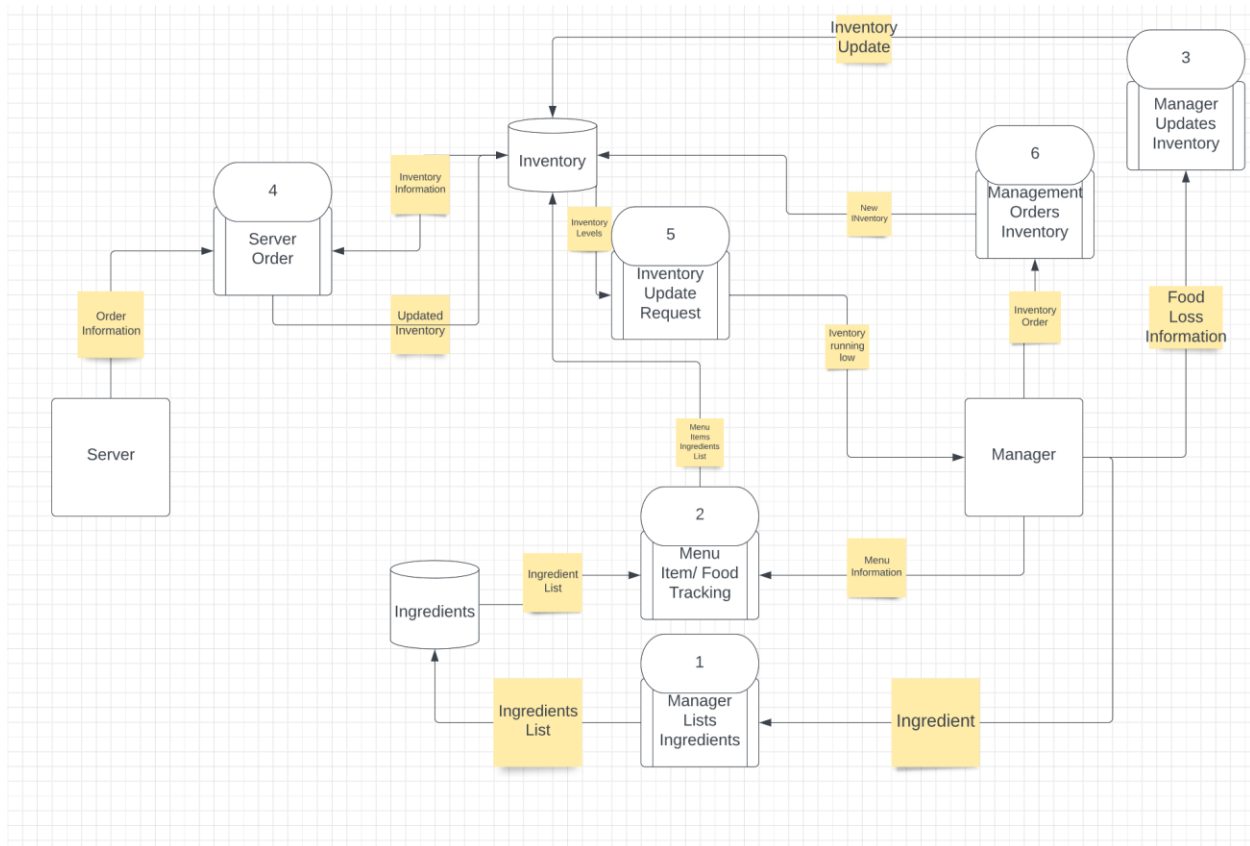
Process Models

Context Diagram



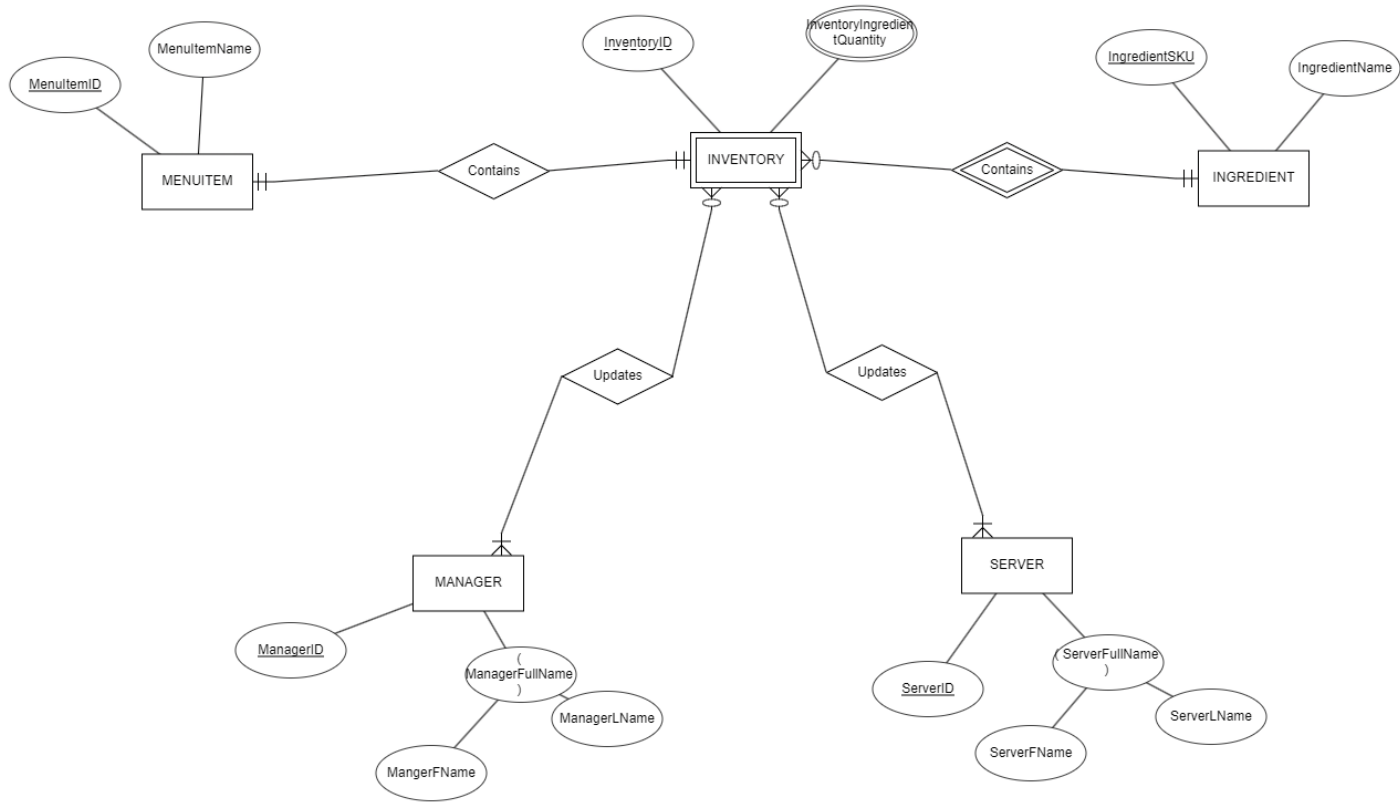
Process Models (*continued.*)

Level 0 Diagram



Data Model

Entity Relationship Diagram





Appendix

Requirements Elicitation Resource:

Questionnaire:

Restaurant System Questionnaire

The following questions will be about your restaurants current system and whether or not a new system would be beneficial.

 jayslice244life@gmail.com (not shared) [Switch account](#) 

1. Does your restaurant have a current system in place for dealing with inventory?

☐ Yes

☐ No

☐ Other: _____

2. If so, are the inventory levels typically accurate on average?

☐ Yes

☐ No

☐ No current system

3. For those who currently don't have a system in place for inventory, do you think a new system would be beneficial to the business?

☐ Yes

☐ No

☐ N/A

4. Adding on to the question above, would you prefer the system to automatically update the inventory levels once you place the order in the system?

☐ Yes

☐ No

☐ N/A

5. For managers, would you be interested in a system that would update inventory levels automatically once you place an order and the order arrives?

☐ Yes

☐ No

☐ N/A


Thanks for taking time to complete the questionnaire!

[Submit](#) [Clear form](#)

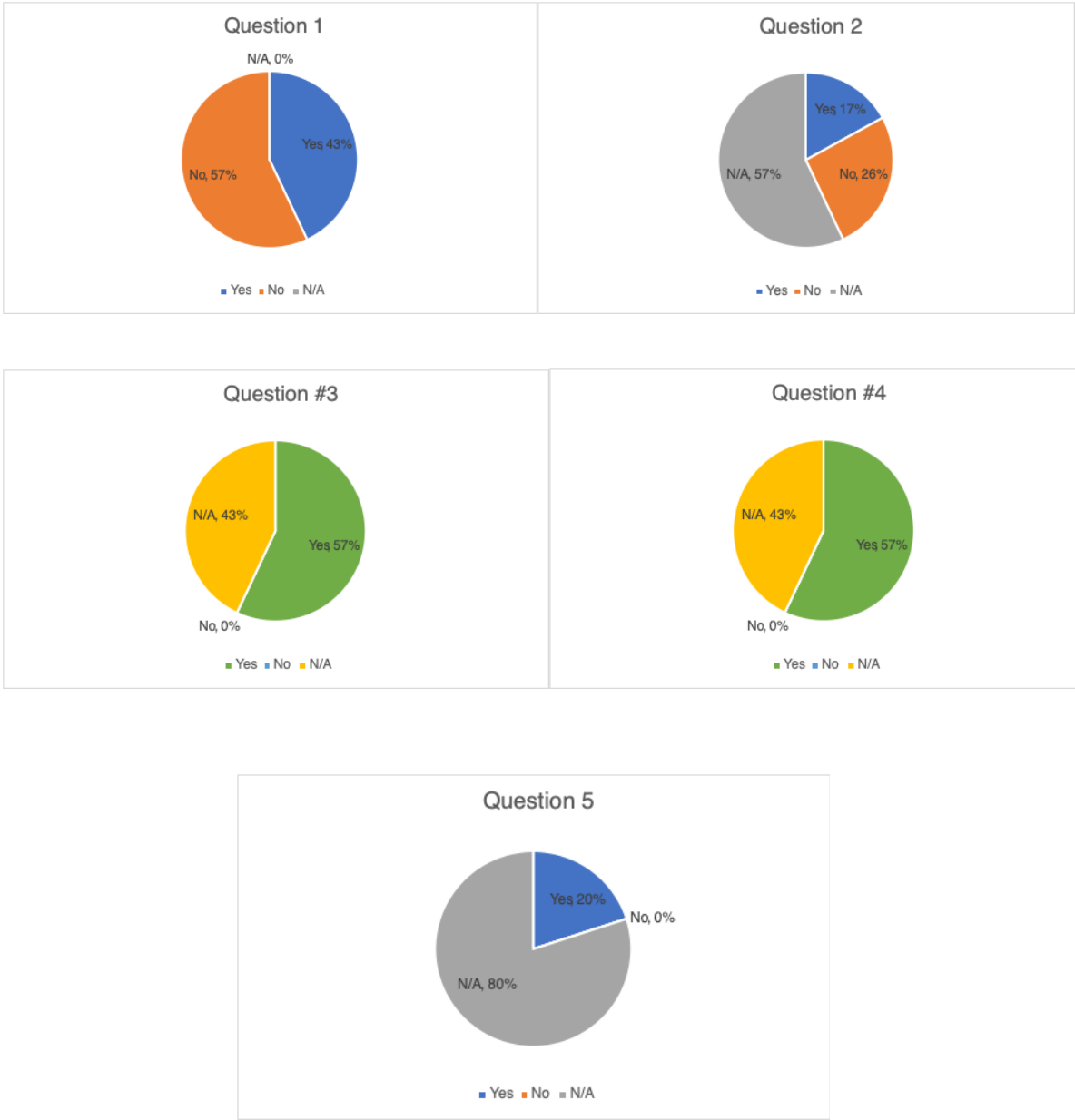
Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Privacy Policy](#)

Google Forms



Questionnaire Results:



Cost Benefit Analysis for Feasibility Study:

Krusty Krab Inventory Management System Simple Cost-Benefit Analysis						
	2022	2023	2024	2025	2026	Total
Benefits						
Reduced Inventory Costs		50,000	50,000	50,000	50,000	200,000
Reduced Inventory Checking Labor		30,000	32,000	34,000	36,000	132,000
Increased Sales		20,000	25,000	35,000	47,500	127,500
Reduced Waste/Spoilage		15,000	15,000	15,000	15,000	60,000
Reduced Theft		1,500	1,500	1,500	1,500	6,000
Total Benefits		116,500	123,500	135,500	150,000	525,500
Development Costs						
1 Server	40,000	0	0	0	0	40,000
Software Licenses	14,000	0	0	0	0	14,000
Server Software	6,000	0	0	0	0	6,000
Development Labor	90,000	0	0	0	0	90,000
Total Development Costs	150,000	0	0	0	0	150,000
Operational Costs						
Hardware		15,000	15,000	15,000	15,000	60,000
Software		5,000	5,000	5,000	5,000	20,000
Operational Labor		40,000	42,000	44,000	46,000	172,000
Total Operational Costs		60,000	62,000	64,000	66,000	252,000
Total Costs	150,000	60,000	62,000	64,000	66,000	252,000
Total Benefits - Total Costs	(150,000)	56,500	61,500	71,500	84,000	273,500
Cumulative Net Cash Flow	(150,000)	(93,500)	(32,000)	39,500	123,500	
Return on Investment	109%	(\$273,500/\$252,000)				
Break- Even Point	2.45 years	2+(((\$71,500-\$39,500)/\$71,500)				

Cost Benefit Analysis for Feasibility Study (*continued*):

*Present Values calculated using a 10% discount rate