



**Business Analysis for Practitioners
- Requirements Elicitation and
Analysis (Domain 3)**

COURSE STRUCTURE

**Introduction to
Business
Analysis
Module 1**

**Needs
Assessment
Module 2**

**Business
Analysis
Planning
Module 3**

**Requirements
Elicitation and
Analysis
Module 4**

**Traceability and
Monitoring
Module 5**

**Solution
Evaluation
Module 6**

COURSE OBJECTIVE

At the end of this course, you will understand what business analysis is all about, why it is essential to the success of any project and how to perform it on your projects...



Business Analysis for Practitioners

MODULE 4



MODULE OBJECTIVE

What is Requirements Elicitation in Business Analysis?

How do we elicit requirements from our stakeholders?

How do we analyse the results?



REQUIREMENTS ELICITATION AND ANALYSIS

Requirement elicitation involves:

- Drawing out information from stakeholders and other sources about the causes of the business problem or the reasons for addressing a current opportunity**
- Drawing out information that will be used to derive sufficient level of requirements to enable solution development and implementation**

REQUIREMENTS ELICITATION AND ANALYSIS

The BA focuses highly on requirements management because this is fundamental to project success – scope, cost and schedule

Ever been on a project team that failed?

If adequate requirements elicitation had been done, do you think the odds for that failure would have been greatly reduced?



Importance of Eliciting Information

Information is not only elicited to generate requirements or answer questions from the solution team, but the information becomes the basis to effectively complete other business analyst tasks, such as:

- Support executive decision making.
- Apply influence successfully.
- Assist in negotiation or mediation.
- Resolve conflict.
- Define problems.

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



Develop the elicitation plan

Prepare for elicitation

Conduct elicitation activities

Elicitation techniques

Document outputs from elicitation activities

Complete elicitation

Elicitation challenges

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



Plan for Elicitation

- **Develop the Elicitation Plan**
(Finding Information, Techniques for Eliciting Information and Sequencing the Elicitation Activities)

Prepare for Elicitation

- **Determine the Objectives**
- **Determine the Participants**
- **Determine the Questions for the Session**

Develop the Elicitation Plan

Some of the elements in an elicitation plan include but are not limited to:

- **What information to elicit.** What does the business analyst need to know in order to define the problem, solve the problem, or answer the question?
- **Where to find that information.** Where is that information located: in what document, from what source, in whose mind? Identify who has the information or where it is located.
- **How to obtain the information.** What method will be used to acquire the information from the source?
- **Sequencing the Elicitation Activities.** In what order should the elicitation activities be sequenced to acquire the needed information?.

Example of Completed Elicitation Plan

What Information	Source	Method	Sequence
How many employees will be moved?	HR	Interview	2
What is the physical layout of both buildings?	Building plans Facilities	Document analysis Interview	1
What equipment will be moved?	IT Facilities Executive VP	Meeting	4
Shall we move in all at once or in a phased approach?	HR Executive VP	Interviews	3
Is it possible to move people in before the building is completely done?	Legal HR Facilities Building contractor	Meeting	5

Prepare for Elicitation

Elicitation preparation is the planning performed to conduct an effective elicitation session. Preparation notes can be used to measure the progress achieved in a session against what was planned to be achieved and can be used to adjust expectations for future sessions.

1. Determine the Objectives
2. Determine the Participants
3. Determine the Questions for the Session

Example of Preparation Notes for an Elicitation Session

Objective

Come to agreement on what equipment will be moved

Participants

Director of IT Operations, Director of Facilities, HR Manager, Director of Service Support

Questions

- Will there be any new equipment purchased to replace old equipment?
- If there is new equipment, will it be delivered to the new or old address?
- Will the move of the computer equipment be the responsibility of IT or the general moving company?
- Will the employees be responsible for any part of the move, for example, moving personal items, or will these be boxed up and moved by the moving company?
- What insurance will cover the equipment during the move?
- Is there a specific order in which the equipment is required to be moved?

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



Conduct Elicitation Activities

- **Introduction**
- **Body** (Types of Questions, How to Ask the “Right” Questions and Listening)
- **Close**
- **Follow-Up**

Conduct Elicitation Activities

There are four stages during the actual elicitation activity in which information is gathered:

- **Introduction.** The introduction sets the stage, sets the pace, and establishes the overall purpose for the elicitation session.
- **Body.** The body is where the questions are asked and the answers are given.
- **Close.** The close provides a graceful termination to the particular session.
- **Follow-Up.** The follow-up is where the information is consolidated and confirmed with the participants.

Types of Questions

The types of questions that can be asked are as follows:

- **Open-ended question.** A question that allows the respondents to answer in any way they desire.
- **Closed-ended question.** A question that calls for a response from a limited list of answer choices. Types of closed-ended questions are forced choice, limited choice, and confirmation.
- **Contextual question.** A question that requires an answer regarding the subject at hand; namely, the problem domain or the proposed solutions.
- **Context-free question.** A question that may be asked in any situation. Context-free questions are also used as lead-ins to obtain information to define the solution.

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

Elicitation techniques

- **Brainstorming**
- **Document Analysis**
- **Facilitated Workshops**
- **Focus Groups**
- **Interviews**
- **Observation**
- **Prototyping**
- **Questionnaires and Surveys**

REQUIREMENTS ELICITATION AND ANALYSIS

Elicitation techniques include:

Brainstorming

**Facilitated
workshops**

Interviews

Observations

**Prototyping
(Storyboarding
& wireframes)**

Questionnaires

**Document
analysis**

Surveys

Focus groups

Interviews

There are two basic types of interviews:

- **Structured interviews.** Begin with a list of prepared questions with the goal of asking and obtaining answers to every question on the list or within the allotted time.
- **Unstructured interviews.** Begin with a list of prepared questions, but the only question that is definitely asked is the first. After that, the subsequent questions are based on the answers to the previous questions.

Interviews may be conducted :

- **Synchronous Interviews.** These interviews are performed live or in real time. These can be conducted in a number of ways, such as face-to-face where the business analyst and the interviewee are in the same room, or they can be conducted over the telephone, with video conferencing, internet collaboration tools, etc. The key is that the interview is being conducted with the interviewee and interviewer at the same time.
- **Asynchronous interviews.** These interviews are not conducted in real time; the business analyst or interviewer is not involved in the interview at the same time as the interviewee.

Observation

Observation is a technique that provides a direct way of viewing people in their environment to see how they perform their jobs or tasks and carry out processes. It is particularly helpful for detailed processes when people who use the product have difficulty or are reluctant to articulate their requirements.

There are four types of observation; each is used in a different situation:

- **Passive observation**
- **Active observation.**
- **Participatory observation.**
- **Simulation.**

Prototyping

Prototyping is a method of obtaining early feedback on requirements by providing a working model of the expected product before building it.

A prototype can be a mockup of the real result as in an architectural model, or it can be an early version of the product itself.

The key element to prototyping is the iterative process of creating the prototype, reviewing it with the pertinent stakeholders, making adjustments and modifications to the prototype, and reviewing it again. This process continues until all parties agree that the prototype has provided the needed information to the solution team.

Prototyping

There are two types of prototypes.: Low-Fidelity and High-Fidelity Prototype

- **Low-fidelity prototype.** Low-fidelity prototypes are completed with a pen and paper, marker and whiteboard, or modeling tool on the computer. Examples of low-fidelity prototypes include:

- Wireframes,
- Mockups of interface screens or reports,
- Architectural renderings of a building,
- Floor plans,
- Sketches of a new product, and
- Any design that is evolving.

Prototyping

- **High-fidelity prototyping.** High-fidelity prototypes create a representation of the final finished product

There are two types of high-fidelity prototypes: throwaway and evolutionary.

○ **Throwaway prototypes** are discarded once the interface has been confirmed. This is similar to the product prototypes developed by manufacturing companies. The prototype is used to help define the tools and process for manufacturing the product but the prototype itself is not sold.

○ **Evolutionary prototypes** are the actual finished product in process. The first prototype that is reviewed is the earliest workable version of the final product. With each successive prototyping session, more functionality is added or the existing functionality is modified to achieve a higher level of quality.

Note: With agile project teams, evolutionary development is how the product is built. The work is not considered to be a prototype but is an actual slice of the product itself.

Prototyping

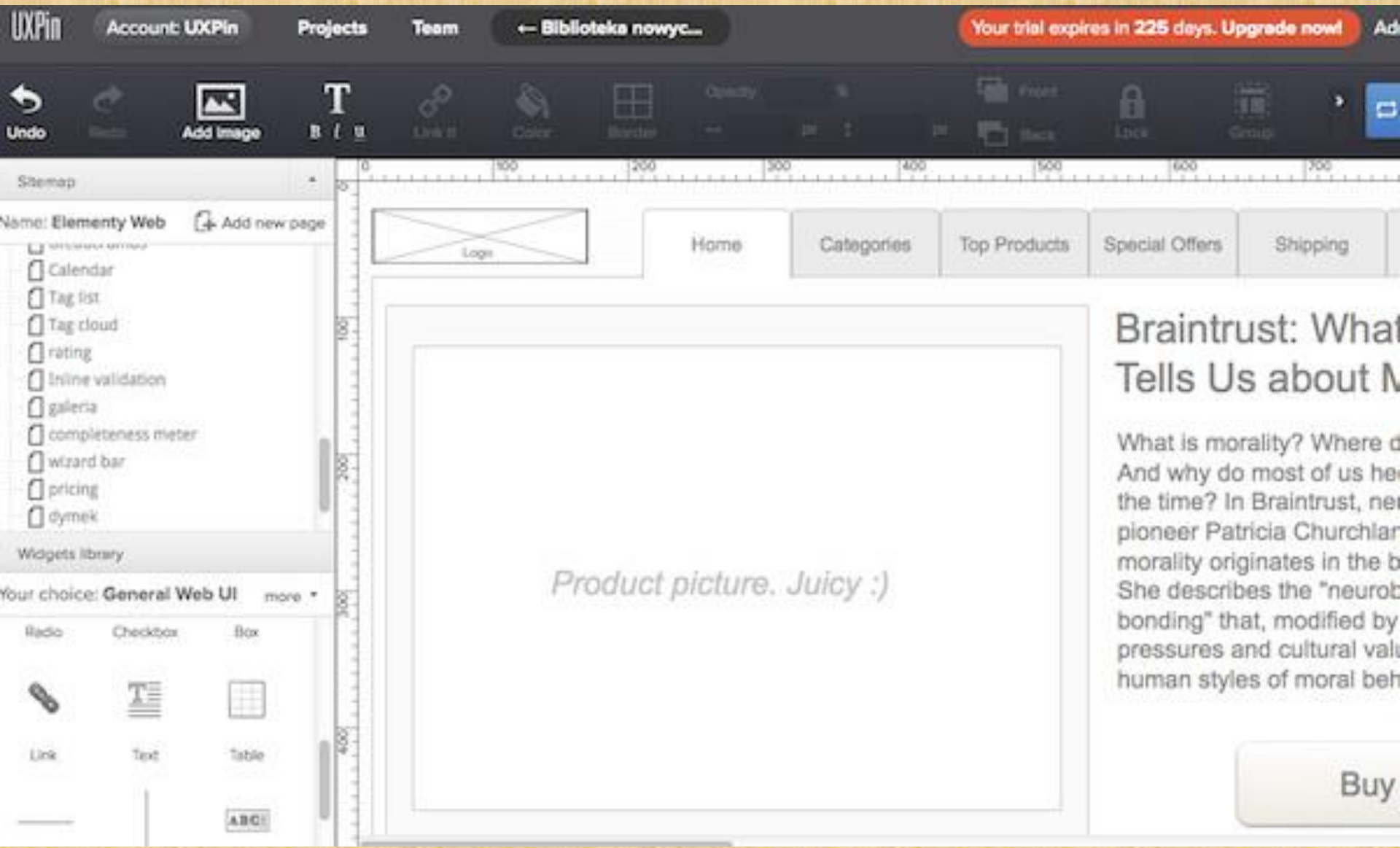
Two examples of prototyping techniques are:

- **Storyboarding.** Storyboarding is a prototyping technique showing sequence or navigation through a series of images or illustrations. Storyboards are graphical representations that represent the sequence of events. Storyboards are typically static and thrown away. Prototypes focus on what the product will look and feel like, while storyboards focus on the experience.
- **Wireframes.** A wireframe is a diagram representing a static blueprint or schematic of a user interface and is used to identify basic functionality. A wireframe separates the look and feel of a site from its function. It presents a stripped-down, simplified version of the page, devoid of distractions. The purpose of the wireframe is to illustrate the flow of specific logical and business functions by identifying all entry and exit points or actions the users will experience.

Difference between Wireframe, Mockup and Prototype

	Fidelity	Cost	Use	General traits
Wireframe	low fidelity	\$	Documentation, quick communication	Sketchy, black, white & grey representation of the interface
Prototype	middle to high fidelity	\$\$\$	User testing, reusable backbone of the interface	Interactive
Mockup	middle to high fidelity	\$\$	Gathering feedback and getting buy-in from stakeholders	Static visualization

Wireframe Example



Mockup Example



Carolina Espresso





[Home](#) | [Menu](#) | [Location](#) | [History of Coffee](#) | [About Us](#) | [Contact](#)

Carolina Espresso

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi a dui eu turpis molestie porta. Maecenas iaculis mattis orci id lacinia. Etiam enim magna, commodo id pellentesque sit amet, feugiat id lectus. Praesent eu ante sapien. Sed elementum ligula ut leo gravida varius. Praesent fringilla vestibulum est, a gravida leo gravida ac.

In egestas pulvinar rhoncus. Mauris est nisi, vestibulum vitae facilisis vel, fringilla a dui. Suspendisse tristique, nisi fermentum pellentesque convallis, augue velit accumsan massa, suscipit gravida mi mauris non diam. Sed sed eros nisi. Sed elementum ligula ut leo gravida varius. Praesent fringilla vestibulum est, a gravida leo gravida ac. Etiam enim



 Latte
  Brewed Coffee
  Frappuccino
  Biscotti
  Cappuccino

News and Events

06/16/09 Carolina Espresso adds two new flavors.

06/14/09 Carolina Espresso is rewarded with the best coffee in Huntersville, NC.

06/13/09 Carolina Espresso Grand Opening. Come check it out.

06/13/09 Carolina Espresso moves to Huntersville, NC with new building.



Customer Reviews

Alex says:

"Carolina Espresso has the best coffee in town. Highly Recommended."

John says:

"Carolina Espresso has friendly service and awesome coffee. You should give it a try. You won't be disappointed."



Prototype Example



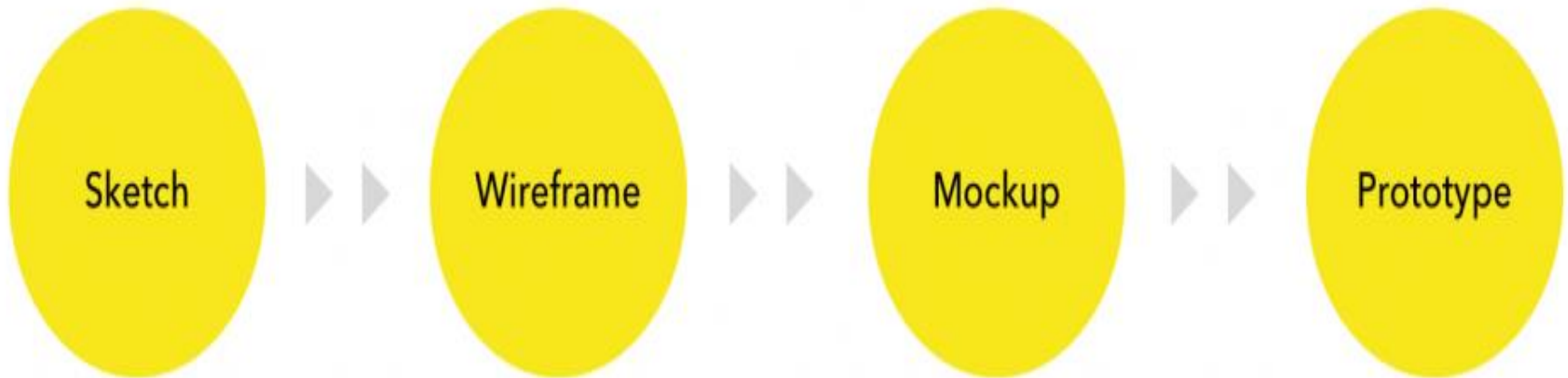
Design Process Example

Process of designing your first app

LOW-FIDELITY
REPRESENTATION

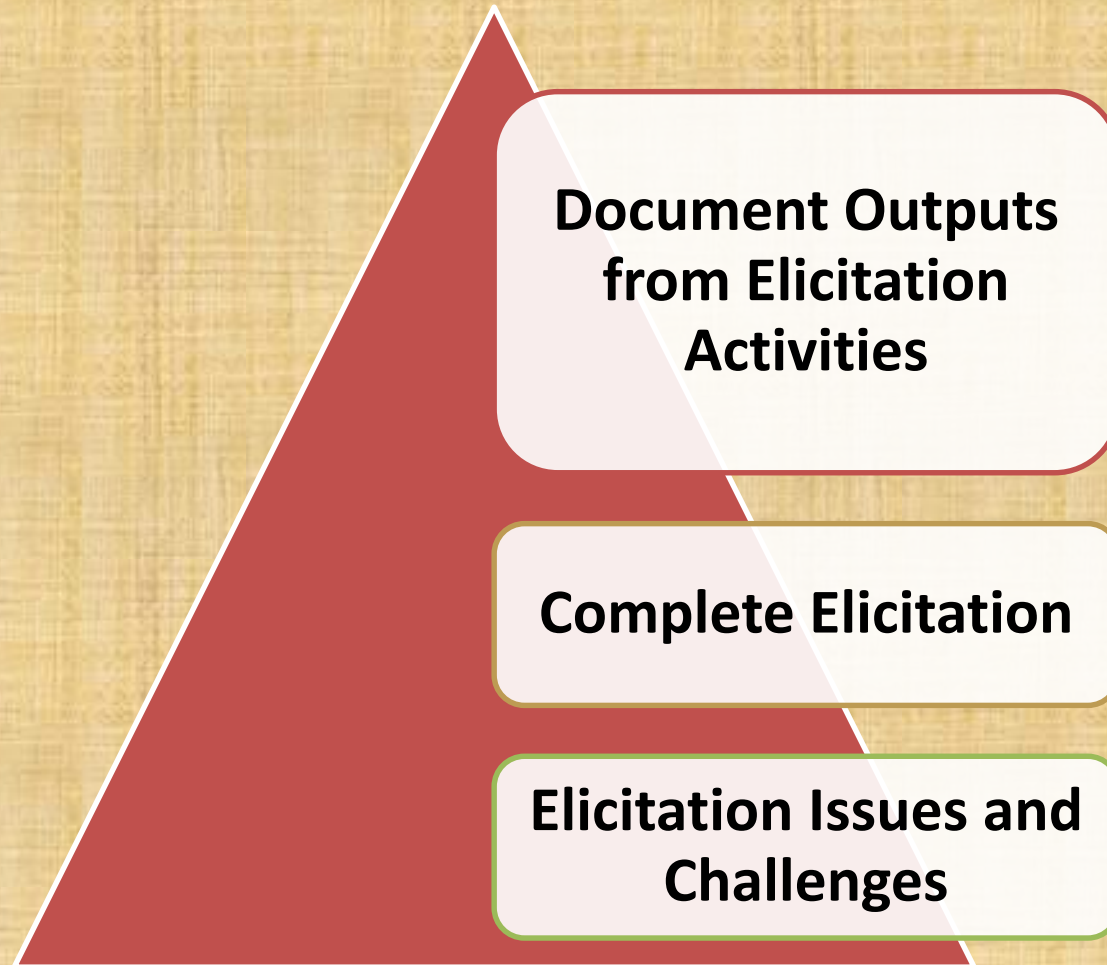
MEDIUM-FIDELITY
REPRESENTATION

HI-FIDELITY
REPRESENTATION



REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

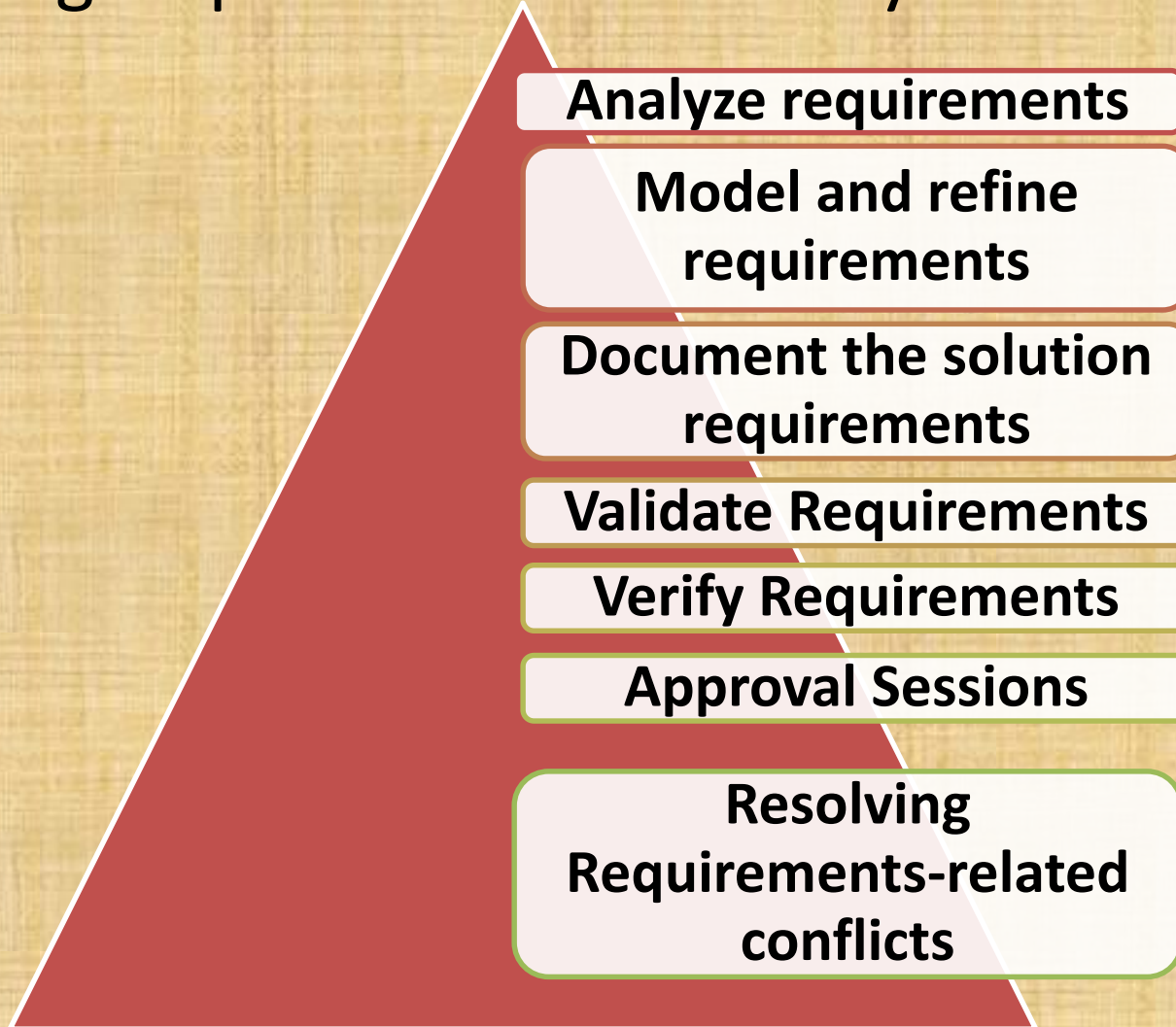


REQUIREMENTS ELICITATION AND ANALYSIS



REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



Analyze Requirements

- **Plan for Analysis** (Analysis Defined, Thinking Ahead about Analysis and What to Analyze)

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

Model and refine requirements

- **Description of Models**
- **Purpose of Models**
- **Categories of Models**
- **Selection of Models**
- **Use of models to refine requirements**
- **Modelling Languages**
- **Scope Models** (Goal Model and Business Objective Model, Ecosystem Map, Context Diagram, Feature Model and Use Case Diagram)



Description and Purpose of Models

Description

The model refers to a visual representation of information, both abstract and specific, that operates under a set of guidelines in order to efficiently arrange and convey a lot of information in a concise manner. In its simplest form, a business analysis model is a structured representation of information.

Purpose

Business analysis models are helpful to find gaps in information and to identify extraneous information. Models provide context to better understand and more clearly convey information. Requirements are modeled and refined to achieve further clarity, correctness, and to elicit additional information to define the details necessary for the product to be built.

Categories of Models

**Model and
refine
requirements**

Models are visual representations of abstract & specific information which help refine requirements

**Scope
models
(e.g.
Ecosystem
maps)**

**Process
models
(e.g.
Process
flow)**

**Rule
models
(e.g.
Decision
tree and
tables)**

**Data models
(e.g. Entity
relationship
diagram)**

**Interface
models
(e.g.
User
interface
flow)**

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

Model and refine requirements

- **Process Models** (Process Flow, Use Case and User Story)
- **Rule Models** (Business Rules Catalogue, Decision Tree and Decision Table)
- **Data Models** (Entity Relationship Diagram, Data Flow Diagrams, Data Dictionary, State Table and State Diagram)
- **Interface Models** (Report Table, System Interface Table, User Interface Flow, Wireframes and Display-Action-Response)

Models Organized by Category

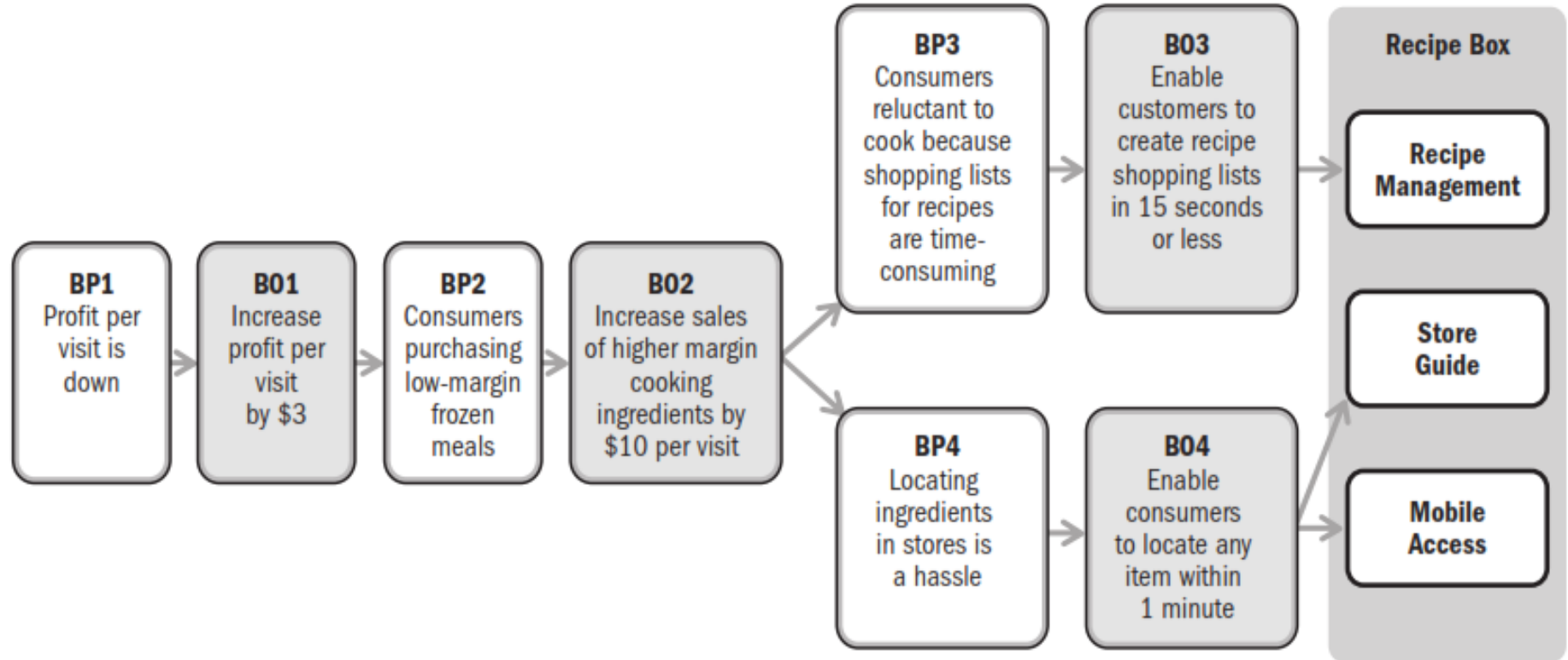
Category	Definition	Example Models
Scope models	Models that structure and organize the features, functions, and boundaries of the business domain being analyzed	<ul style="list-style-type: none"> ▪ Goal and business objectives model ▪ Ecosystem map ▪ Context diagram ▪ Feature model ▪ Organizational chart (described in Business Analysis Planning) ▪ Use case diagram ▪ Decomposition model (described in Business Analysis Planning) ▪ Fishbone diagram (described in Needs Assessment) ▪ Interrelationship diagram (described in Needs Assessment) ▪ SWOT diagram (described in Needs Assessment)
Process models	Models that describe business processes and ways in which stakeholders interact with those processes	<ul style="list-style-type: none"> ▪ Process flow ▪ Use case ▪ User story
Rule models	Models of concepts and behaviors that define or constrain aspects of a business in order to enforce established business policies	<ul style="list-style-type: none"> ▪ Business rules catalog ▪ Decision tree ▪ Decision table
Data models	Models that document the data used in a process or system and its life cycle	<ul style="list-style-type: none"> ▪ Entity relationship diagram ▪ Data flow diagram ▪ Data dictionary ▪ State table ▪ State diagram
Interface models	Models that assist in understanding specific systems and their relationships within a solution	<ul style="list-style-type: none"> ▪ Report table ▪ System interface table ▪ User interface flow ▪ Wireframes ▪ Display-action-response

Modeling Languages and Usage

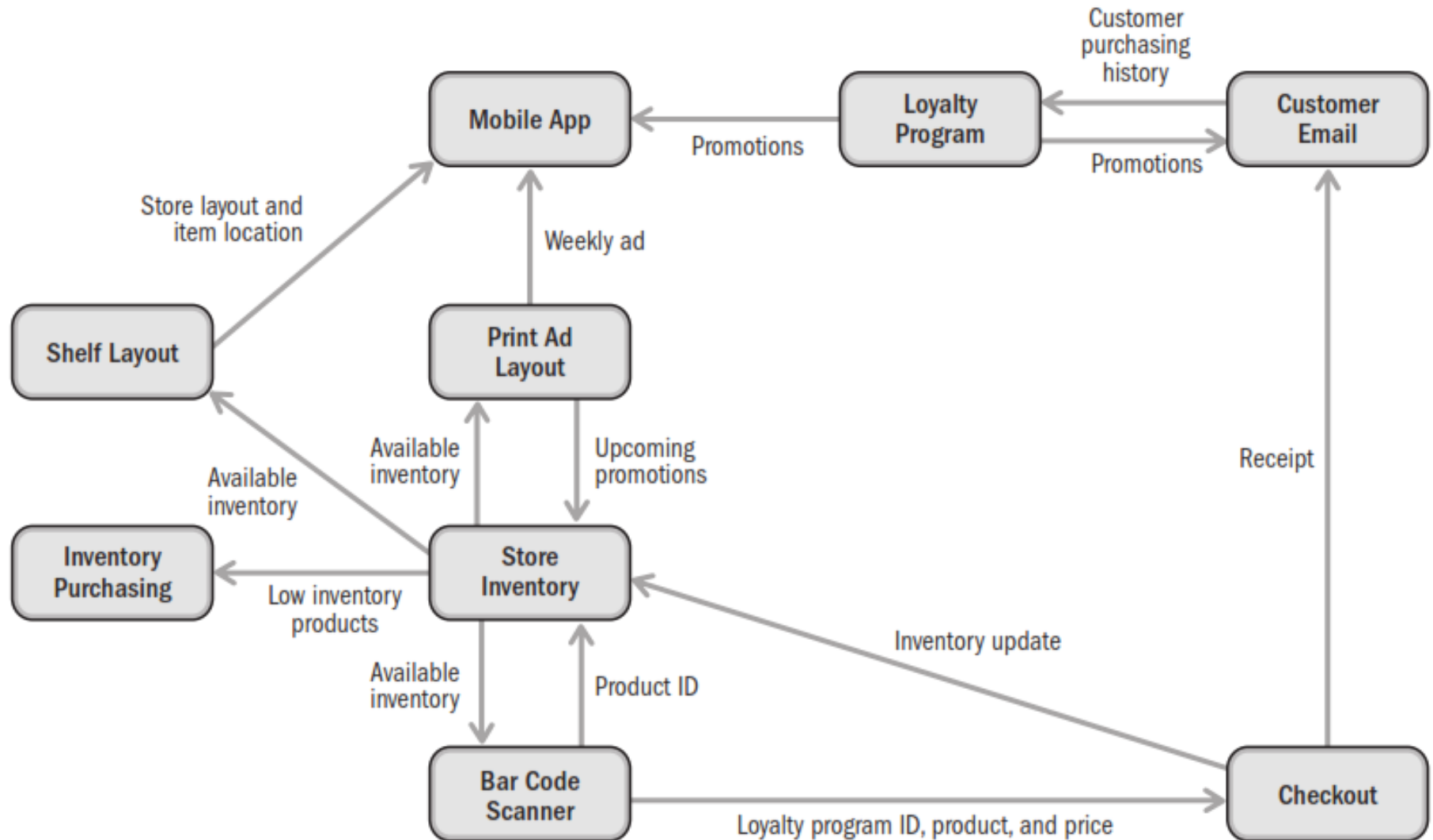
Modeling Language	Overview of Usage
Business process modeling notation (BPMN)	Used to model complex business processes for the purpose of making changes to these processes.
Requirements modeling language (RML)	Used to visually model requirements for easy consumption by all stakeholders, particularly business stakeholders.
System modeling language (SysML)	Used to analyze complex systems and includes a subset of UML.
Unified modeling language (UML)	Primarily used to specify design models but can work well to specify requirements.
Various other modeling languages	Used when a specific modeling language isn't appropriate or not part of the organizational standards. For example, process models are frequently created using ISO-standard flowchart symbols. Data models often use Information Engineering "crow's foot" notation.

Scope Model

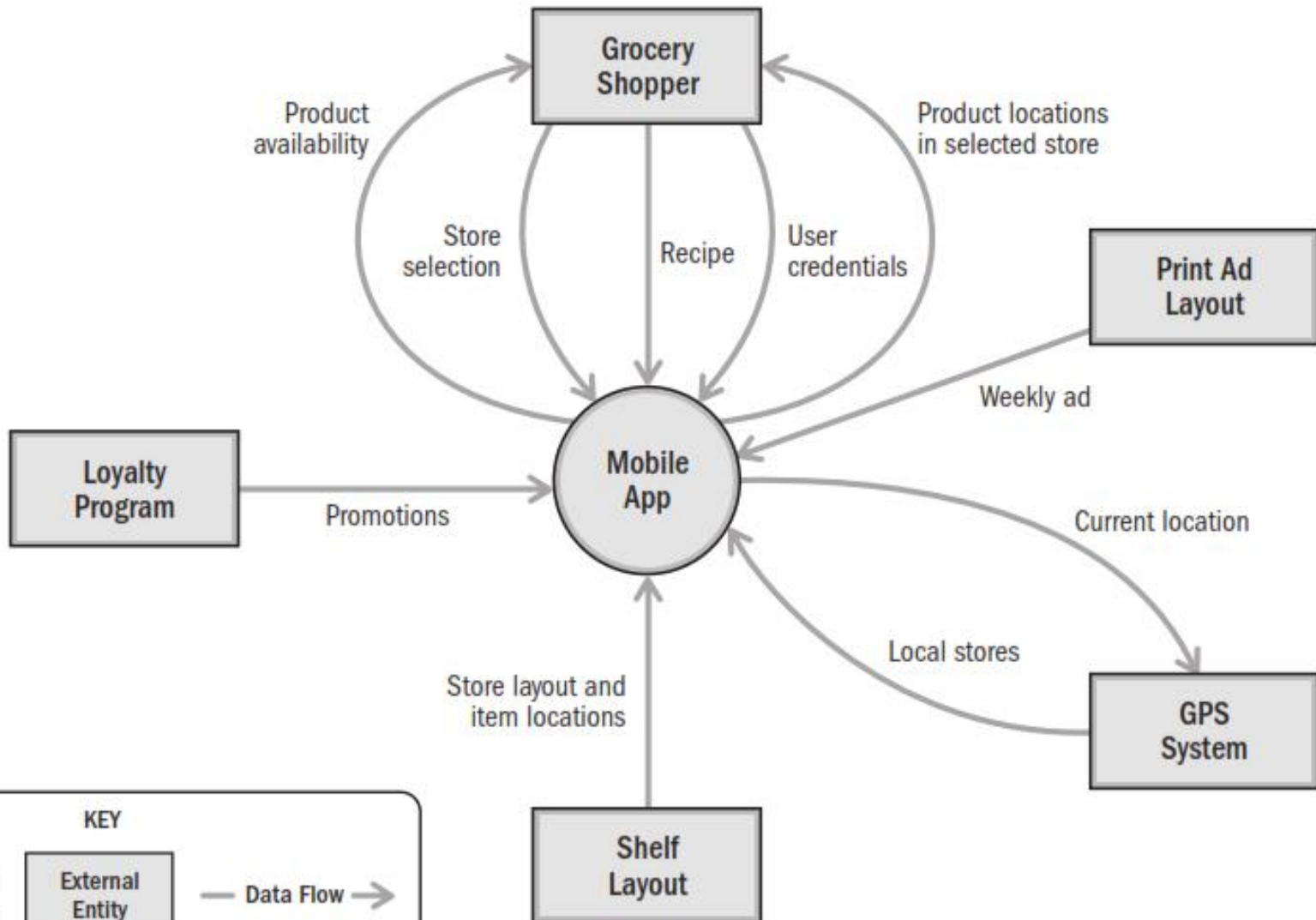
Goal Model and Business Objective Model



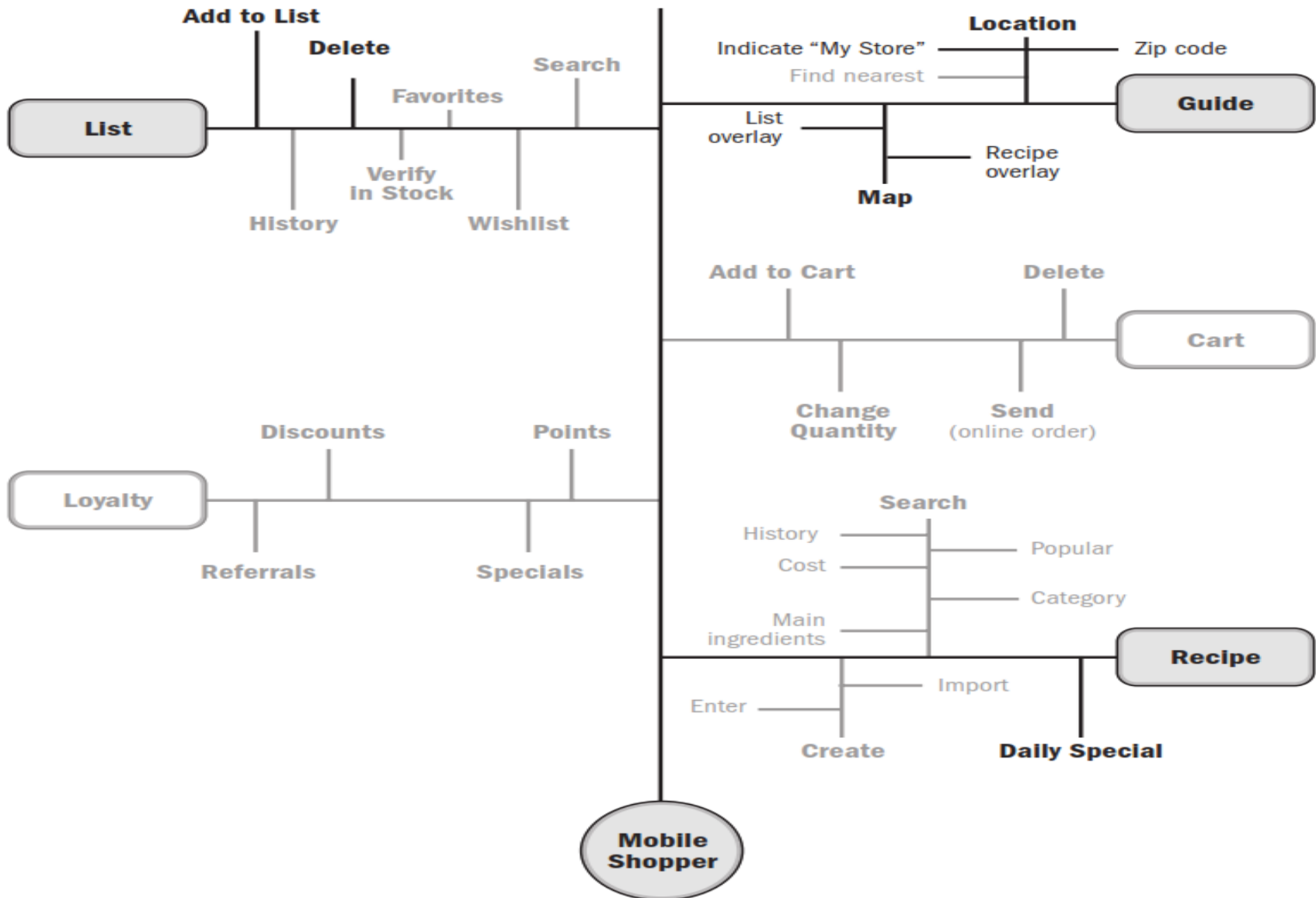
Ecosystem Map



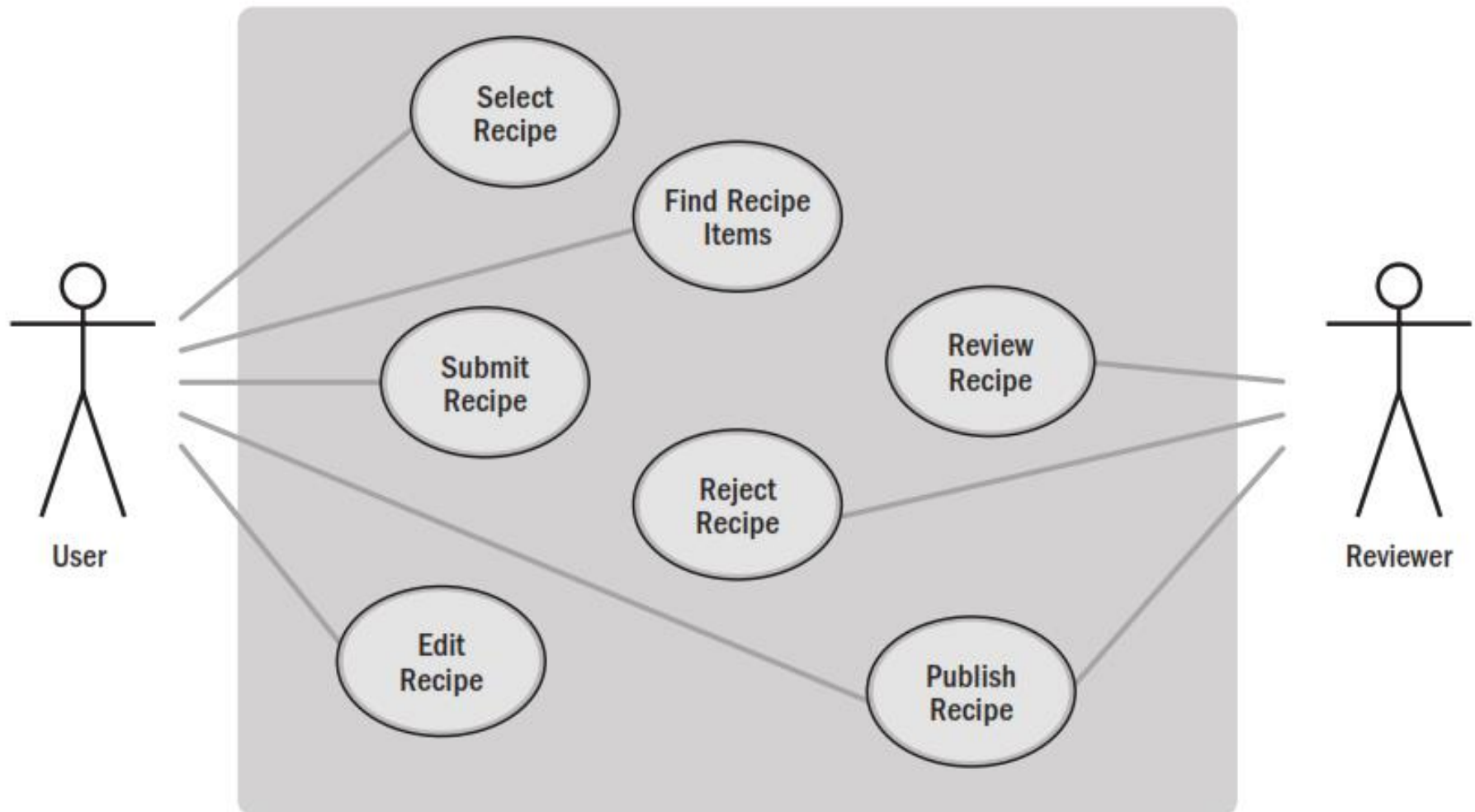
Context Diagram



Feature Model

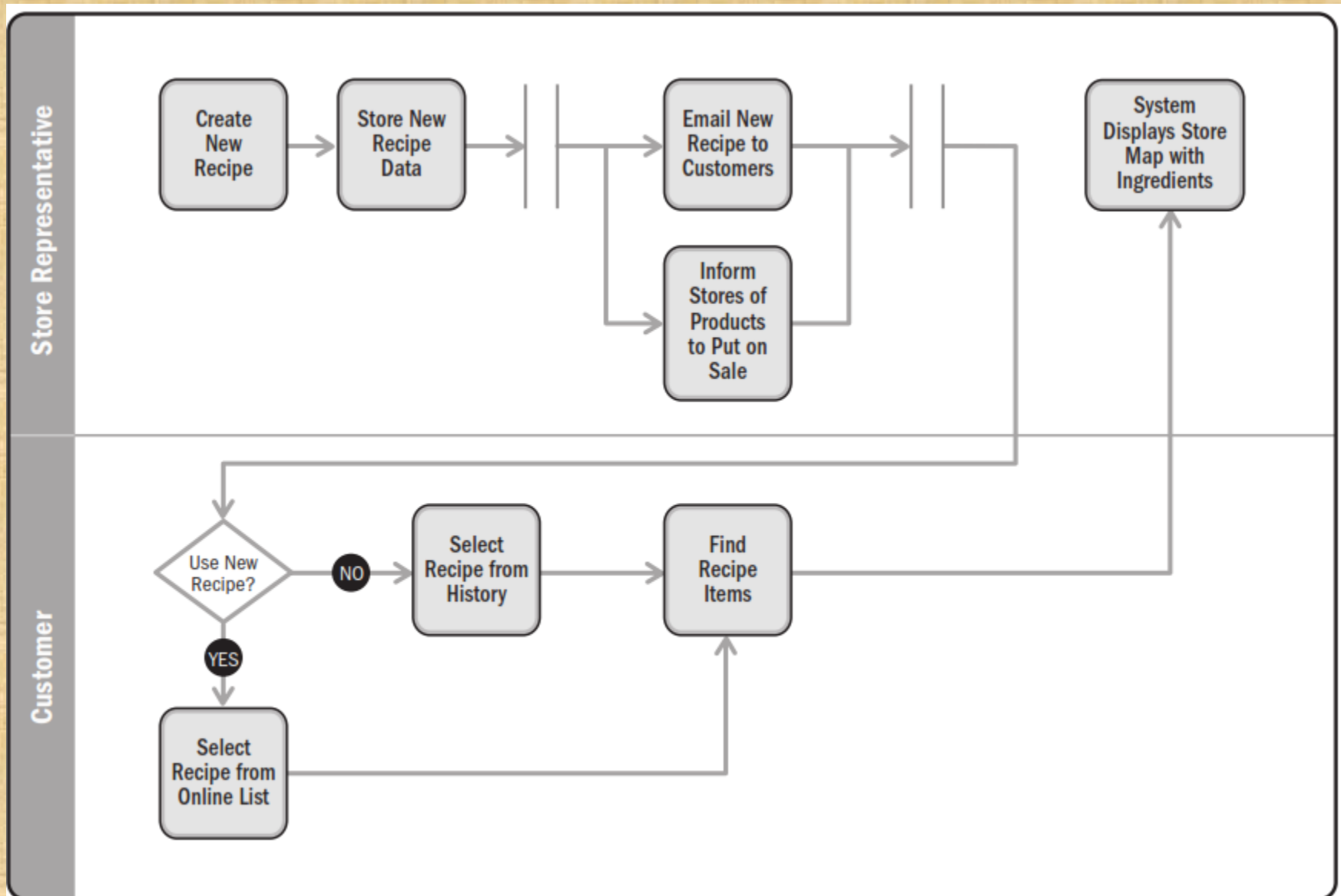


Use Case Diagram



Process Model

Process Flow



Use Case

Name	Find Recipe Items
ID	UC_001
Description	<p>A daily email with the featured recipe is sent to customers who have opted-in to the program. The customer opens the Recipe Box application directly on their iOS or Android device. The customer can choose a past recipe or use the current one.</p> <p>With a recipe chosen, the customer chooses a store or uses a store already indicated a "My Store." A map of the store is displayed with an overlay showing the locations of the items for the recipe.</p>
Actors	User (customer on a mobile app)
Organizational Benefits	Customer visits store to make purchases related to the recipe which are high-margin cooking items.
Triggers	Customer clicks on link in email on mobile device or directly opens application
Preconditions	Recipe Box application opened successfully
Post conditions	Customer views a map of the store with recipe items and locations indicated
Normal Flow	<ol style="list-style-type: none"> 1. System shows short description of the daily recipe 2. Customer selects current recipe or chooses a past recipe 3. Customer chooses to use their local store ("My Store") (see AC1, see AC2) 4. System displays map of the store with the recipe items overlaid as icons 5. Customers can select icons to see item aisle and shelf location (see EX1)
Alternate Flows	<p>AF1—Local store not yet indicated</p> <ol style="list-style-type: none"> 1. System prompts for zip code for stores 2. Customer enters zip code 3. System lists stores nearby to zip code (see EX2) 4. Customer chooses a store to be "My Store" and it is used for recipe map 5. Return to MC step 4 <p>AF2—Local store not used (whether a "My Store" is chosen or not)</p> <ol style="list-style-type: none"> 1. System prompts for zip code for stores 2. Customer enters zip code 3. System lists stores nearby to zip code (see EX2) 4. Customer chooses a store from the list 5. Return to MC step 4
Exception Flows	<p>EX1—Selected store does not have an items for the recipe</p> <ol style="list-style-type: none"> 1. System alerts user about out<>f-stock items 2. Return to MC step 5 <p>EX2—No stores in zip code</p> <ol style="list-style-type: none"> 1. System alerts customer that there are no stores found in zip code and to enter different zip code 2. Return to AC1 or AC2 step 2

User Story

Epic: As a customer, I want to go to the store and easily buy ingredients needed for a recipe.

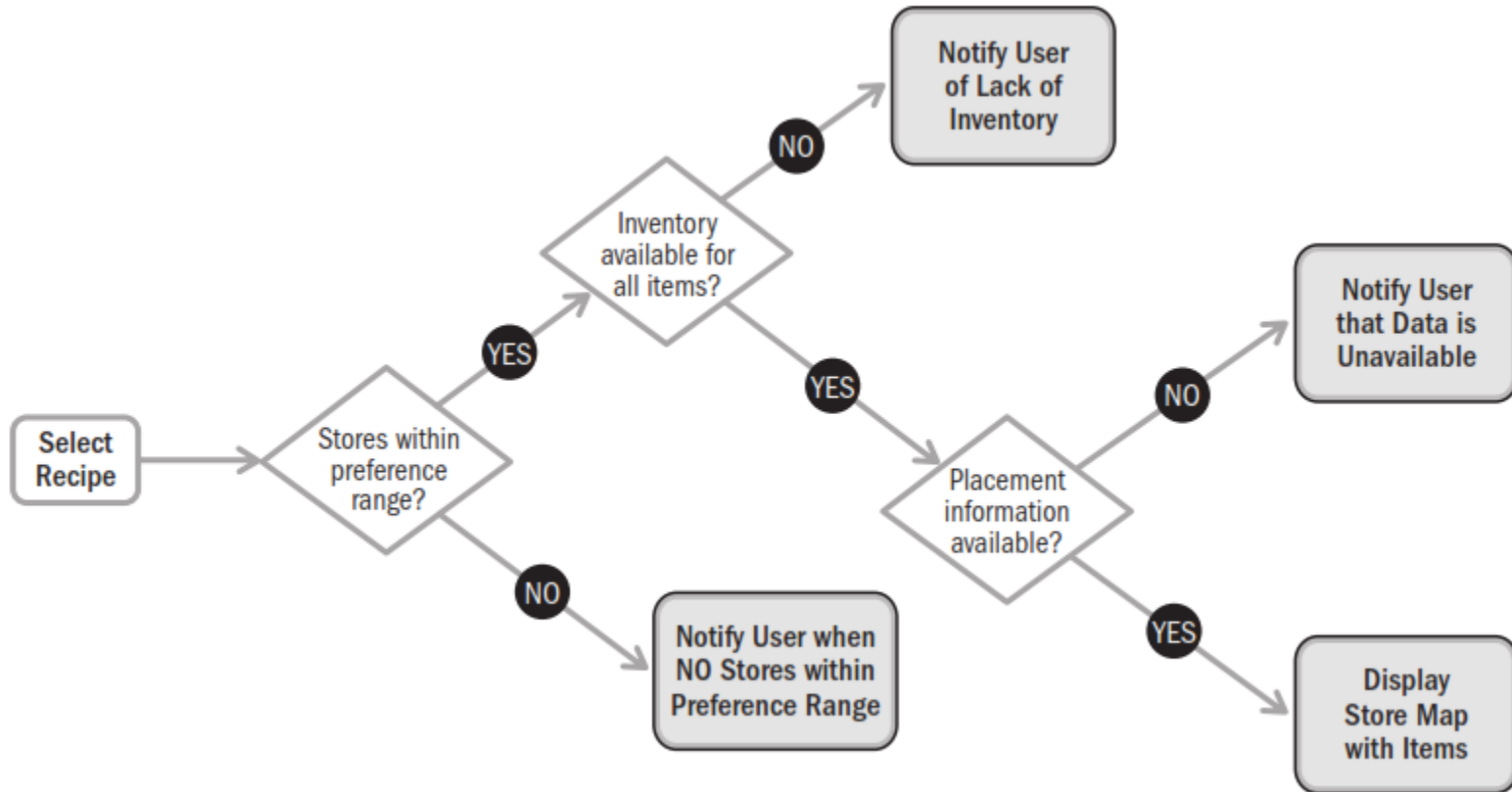
User Story	Acceptance Criteria
As a customer, I want to be able to find past recipes so I can prepare them again.	<ol style="list-style-type: none">1. Customer can search for past recipes2. Search terms can be by recipe name, ingredient, or date3. A search can return 0, 1, or many results4. For one or many results, a user can choose a recipe from the list
As a customer, I want to select a store to purchase recipe ingredients so I can choose a store location that is close to me.	<ol style="list-style-type: none">1. Customer can enter a ZIP Code and get a list of stores in or near that ZIP Code, out to 20 miles from the ZIP Code2. Customer can select a store from the list of stores3. If no stores are found, the customer is informed
As a customer, I want to have a store map of where recipe ingredients are so I don't have to hunt for them.	<ol style="list-style-type: none">1. Customer has selected a recipe and a store2. Customer is shown a map of the store with icons representing the recipe ingredients3. Recipe icons can be selected by click or hover, and will show the ingredient name as well as the aisle and shelf location

Rule Model

Business Rules Catalog

Recipe Box				
BR ID	Business Rule Title	Business Rule Description	Type (fact, computation, constraint, other)	References
BR01	Recipe Email Opt-in	Recipe emails will only be sent to customers who have opted-in and have a valid email address.	Constraint	See corporate email policy
BR02	No PII in Recipe Email	Recipe emails will not contain any personally identifiable information (PII).	Constraint	See corporate email policy
BR03	Ingredients in stock	A new recipe will not be sent when more than 10% of the stores have a restocking status of greater than 24 hours for any of the ingredients	Computation	Will use inventory reporting system

Decision Tree

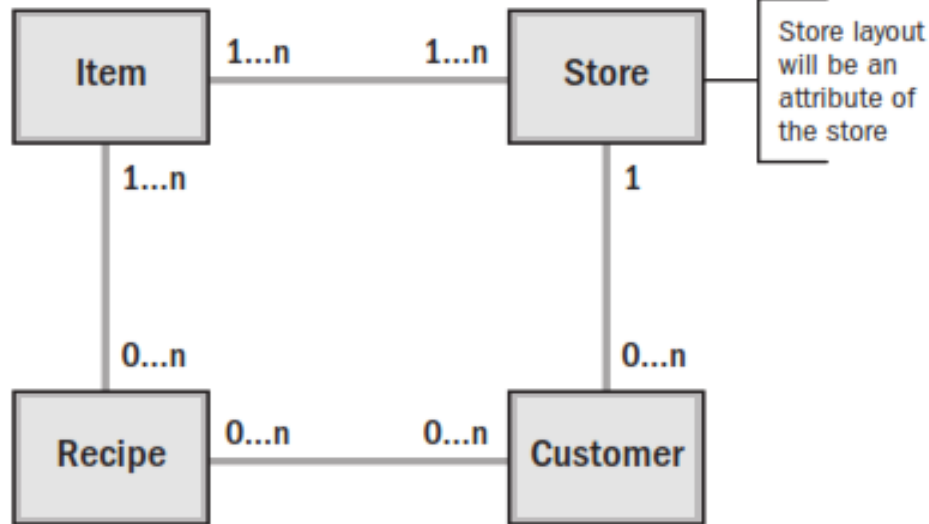


Decision Table

Map Store Decision Table	Rule 1	Rule 2	Rule 3	Rule 4
Conditions				
Stores within preference range	N	Y	Y	Y
Inventory available for all items	-	N	Y	Y
Placement information available	-	-	N	Y
Outcomes				
Notify user when NO stores within preference range	X	-	-	-
Notify user of lack of inventory	-	X	-	-
Notify user that data is unavailable	-	-	X	-
Display store map with items	-	-	-	X

Data Model

Entity Relationship Diagram



Key

1

Single cardinality

0...n

None up to many cardinality

0...1

None or single cardinality

1...n

Single up to many cardinality



Relationship

Business data object

Optional One



Optional Many



Mandatory One



Mandatory Many



Key

1

Single cardinality

0...n

None up to many cardinality

0...1

None or single cardinality

1...n

Single up to many cardinality

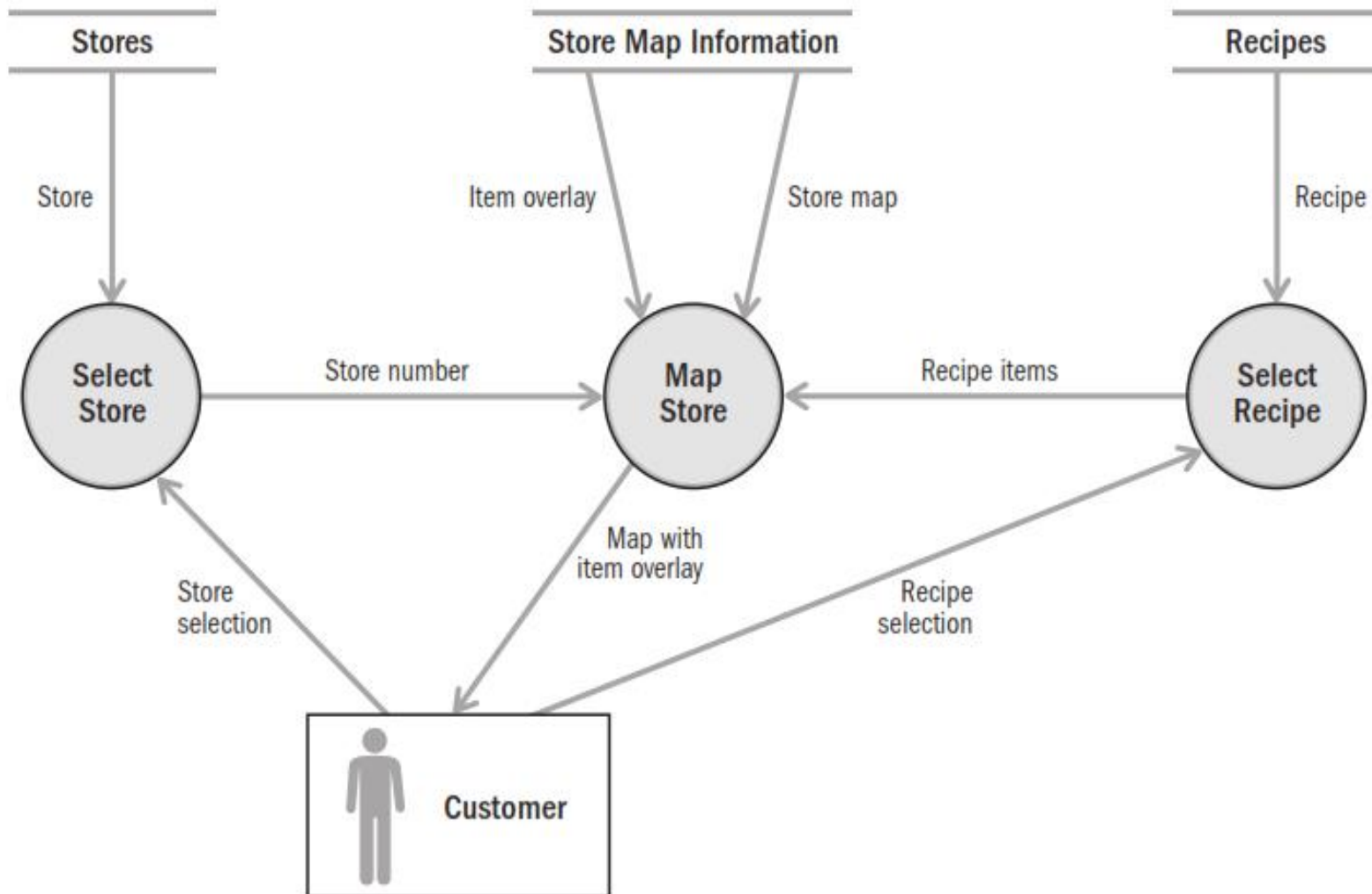


Relationship

Business data object

Crows' Foot and 1 to N notation

Data Flow Diagram



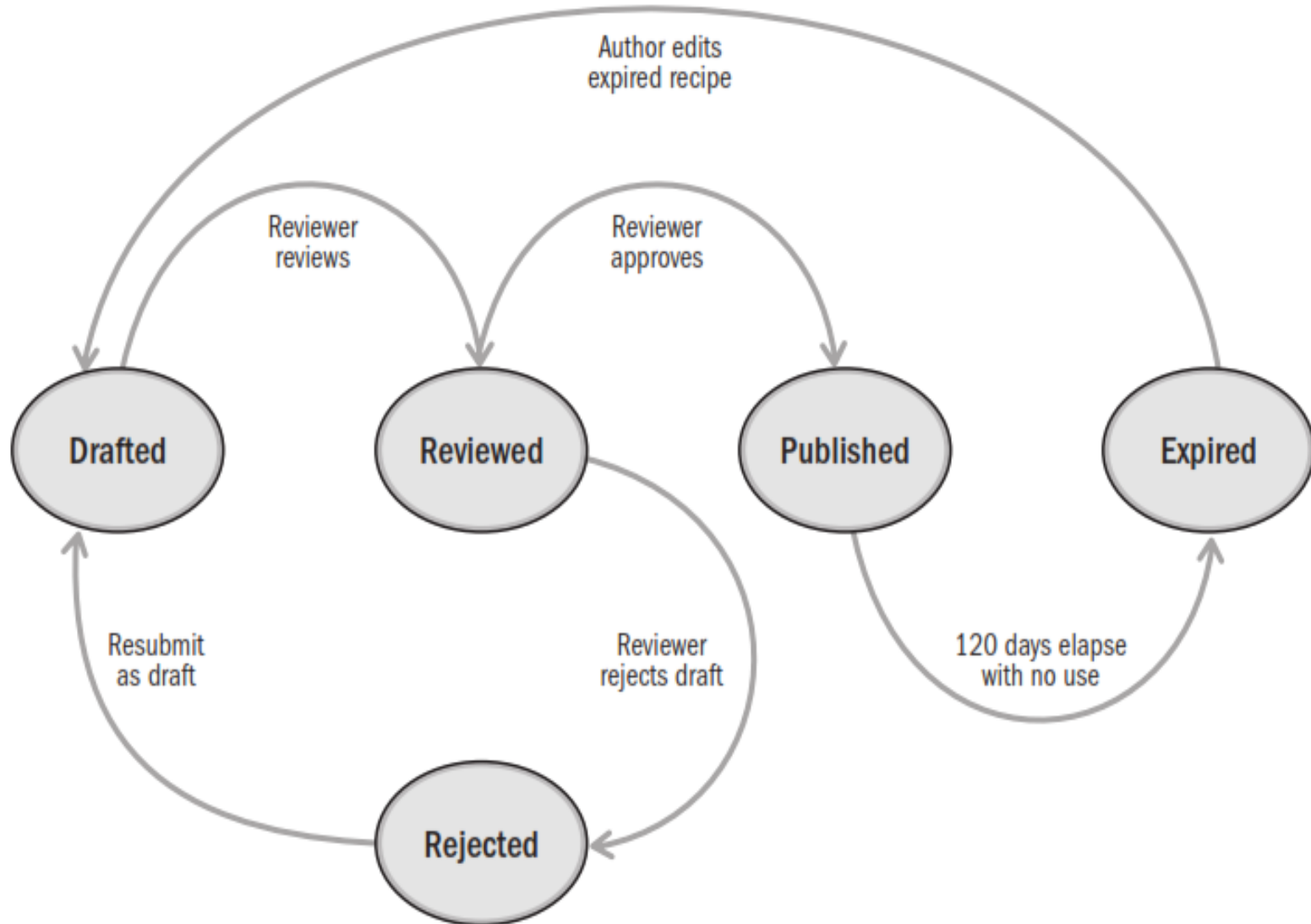
Data Dictionary

ID	Business Data Object	Field Name	Description	Unique Values?	Data Type	Length	Valid Values
RM01	Recipe	RecipeText	Recipe in formatted layout	Y	Alphanumeric	< 1000 characters	Structured text (see REC_FRM01)
RM03	Items	RecipeUPC	UPC collection for a given recipe	Y	Alphanumeric	< 1000 characters	Structured text (see REC_FRM02)
RM02	Store	StoreNum	Store number for map and overlay	Y	Integer	3 numeric characters	Integer > 0 and valid store number
RM04	Store	StoreMap	Graphic of store layout	Y	Graphic	640 x 480 pixels	n/a
RM06	Store	ItemOverlay	Generated array of recipe items, locations, scaled to store layout graphic	Y	Alphanumeric	< 1000 characters	Structured text (see LAYOUT_FRM01)

State Table

		Target State				
		Drafted	Reviewed	Rejected	Published	Expired
Initial State	Drafted	X	Reviewer reviews	X	X	X
	Reviewed	X	X	Reviewer rejects draft	Reviewer approves	X
	Rejected	Resubmit as draft	X	X	X	X
	Published	X	X	X	X	120 days elapse with no use
	Expired	Author edits expired recipe	X	X	X	X

State Diagram



Interface Model

Report Prototype

Mobile App Usage Report (MAUR)

Reporting Period: September 07, 2014-September 13, 2014

Number of App Downloads	Number of Recipe Uploads	Store ID	Region	Number of Loyalty Program Transactions	Number of Loyalty Program Transactions by Users with Recipe Uploads	Average Number of Products per Transaction for Loyalty with Recipe Transactions	Average Number of Products per Transaction for Loyalty without Recipe Transactions
10,732	6,907	0	0	78,563	4,576	34	10
		23548	Central Texas	10,723	2,469	32	17
		54721	Central Texas	6,093	1,098	33	15
		56098	Central Texas	15,497	543	30	12

Top-Level Elements in a Report Table

Top-Level Elements	Element	Description
	Unique ID	REP_MBL001
	Name	Mobile App Usage Report (MAUR)
	Description	The MAUR contains the total numbers of new app downloads, recipe uploads, and frequency of visits by app users
	Decisions Made from Report	The MAUR will be used to determine if the Recipe Box feature is being utilized by mobile app users and if users are actually purchasing the products in stores
	Objective	Business Objective 01 (improve profit per visit)
	Priority	1 of 16
	Functional Area	Sales and Marketing
	Related Reports	Inventory Report (REP_INV001), Sales Report (REP_REV001)
	Report Owner	Marketing Manager
	Report Users	VP of Sales, VP of Marketing, Sales Representatives, Marketing Department, Mobile Services Representatives
	Trigger	MAUR is generated automatically on Monday morning at 2 a.m.
	Frequency	Weekly
	Latency	MAUR is delivered within 2 minutes of being triggered and contains data for the previous week (Monday 12:00 a.m.–Sunday 11:59 p.m.)
	Transaction Volume	Approximately 100,000 transactions are logged weekly
	Security	Viewable by all sales and marketing team members
	Persistence	All settings are saved between report executions
	Visual Format	The Reporting period will be noted above the MAUR Matrix in the form "Month DD, YYYY–Month DD, YYYY" MAUR is a matrix with the following columns: Number of App Downloads, Number of Recipe Uploads, Store ID, Region, Number of Loyalty Program Transactions, Number of Loyalty Program Transactions by Users with Recipe Uploads, Average Number of Products per Transaction for Loyalty with Recipe Transactions, Average Number of Products per Transaction for Loyalty without Recipe Transactions The first row contains metrics for all stores The Store ID and Region will be "O" for the first row
	Delivery Format	The report is emailed in an Excel file
	Interactivity	The MAUR retains normal interactivity available in Excel files
	Drilldowns	N/A

Field Elements in a Report Table

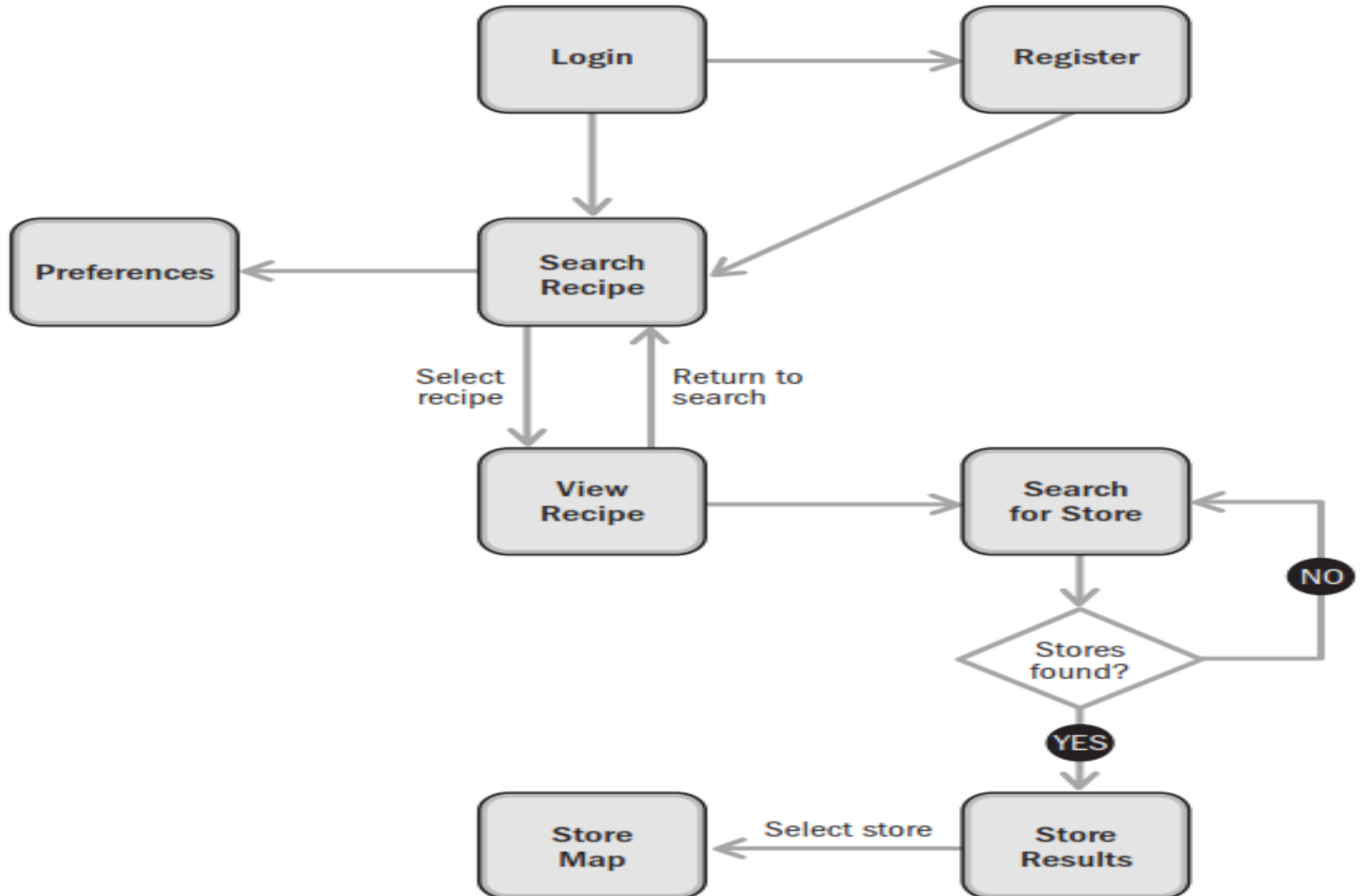
Field Elements	Element	Description
	Filtered By	All columns can be filtered The default setting displays all data
	Grouped By	Rows are grouped by Region
	Sorted By	All columns can be sorted The default setting is to sort by Store ID within Region
	User Input Parameters	N/A
	Group Calculation	N/A
	Calculated Fields	Each calculation is done for the duration of the report period specified: App.DownloadTotal = sum of app downloads Recipe.UploadTotal = sum of recipe uploads Transactions.LoyaltyTotal = sum of transactions using a Loyalty Program ID Transactions.LoyaltyRecipeTotal = sum of transactions using a Loyalty Program ID with recipe upload(s) Transactions.AverageProductsPerLRT = total number of products purchased in Transactions.LoyaltyRecipeTotal divided by Transactions.LoyaltyRecipeTotal Transactions.AverageProductsPerLNRT = total number of products purchased in Transactions.LoyaltyNoRecipeTotal divided by Transactions.LoyaltyNoRecipeTotal
	Displayed Fields	All fields are shown with rounding to the nearest integer App.DownloadTotal Recipe.UploadTotal Store.StoreNum Store.Region Transactions.LoyaltyTotal Transactions.LoyaltyRecipeTotal Transactions.AverageProductsPerLRT Transactions.AverageProductsPerLNRT

System Interface Table

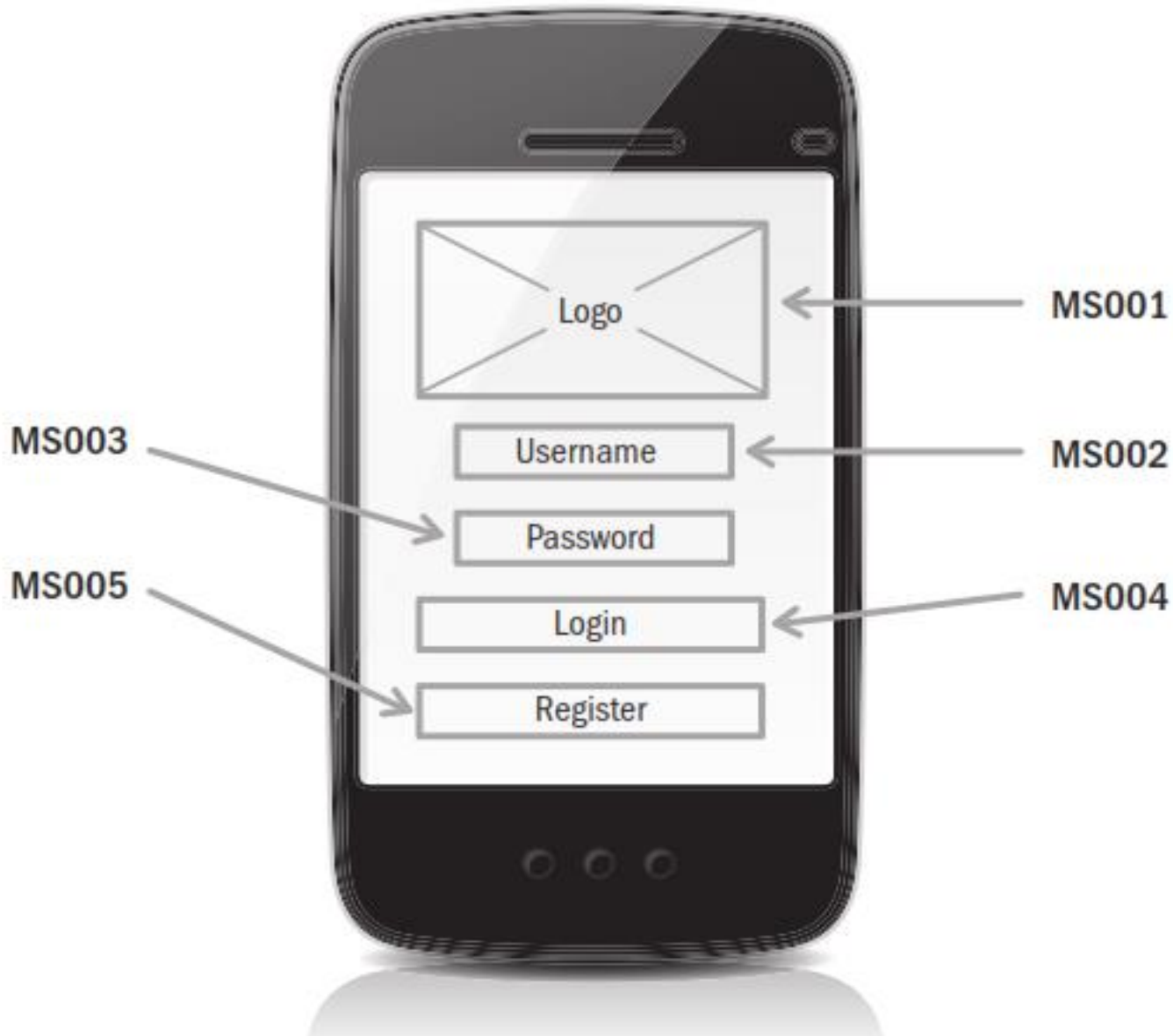
System Interface	
Source	Store
Target	Customer mobile device
ID	MS_01
Description	Passes store and recipe information to mobile application
Frequency	Daily
Volume	<1000 recipes
Security Constraints	None
Error Handling	See Sync_Store_Recipe_System_Flow

Interface Objects			
Object	Field	Data Dictionary ID	Validation Rule
Store	StoreNum	RM02	Positive integer
Store	StoreMap	RM04	JPEG image
Recipe	RecipeText	RM01	Well-formed structured text (ref data dictionary)
Store	ItemOverlay	RM06	Well-formed structured text (ref data dictionary)

User Interface Flow



Wireframe



Display-Action Response

UI Element: Password Field

UI Element Description

ID	MS003
Description	A field for the user to enter their password

UI Element Displays

Precondition	Display
On login screen	Display field
No text entered	Grayed out password
Text entered	Privacy dots are displayed per character

UI Element Behaviors

Precondition	Action	Response
On login screen	Select field	Keyboard appears
Text entered	Enter additional text	Privacy dots per character appear

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

Document the solution requirements

- Why Documentation is Important
- Business Requirements Document
 - The Solution Documentation
(Requirements and Categorization)
- Requirements Specification (Documenting Assumptions and Documenting Constraints)
- _ Guidelines for Writing Requirements
(Functional Requirements)



Why Documentation is Important

Documented requirements serve a multitude of purposes, such as the following:

- Baseline to validate the stakeholder needs;
- Baseline definition of the solution for the business problem or opportunity;
- Primary input to the design team, the developers, testers, and quality assurance;
- Basis for user manuals and other documentation whether written or embedded;
- Supporting detail for contractual agreements, when applicable (e.g., the requirements are a core input to a statement of work (SOW) for a request for proposal (RFP));
- Starting point for the evolution of the solution;
- Foundation for reusability by other project teams who need an understanding of the project details while it is in process or after implementation; and
- Baseline for an audit of regulated industries and high-risk projects that are required to have documented requirements.



Business Requirement and Solution Document

Business requirements are the goals, objectives, and higher-level needs of the organization and provide the rationale for a new project. Business requirements recognize what is critical to the business and why it is critical before defining a solution.

Solution documentation is the documentation that is comprised of the features, functions, and characteristics of the product or service that will be built to meet the business and stakeholder requirements. It serves as the blueprint for the product that the solution team is being asked to build.

The solution documentation may be rendered in any number of forms. Some common forms include:

- Requirements document, which may be a business requirements document and/or a functional requirements specification and/or a system or software requirements specification, etc.;
- Deck of user stories written;
- Set of use cases with accompanying nonfunctional requirements; or
- List of items on a product backlog.



Product and Project Requirement

While product requirements describe what is being built or the outcome of the project or solution to the business problem, project requirements describe the constraints and necessities for successful completion of the project.

For example, product requirements describe the length and width of the sidewalk to be constructed in front of a building, along with such aspects as color and texture. The project requirements for laying the sidewalk could include the number of laborers required, qualifications of the laborers to handle the equipment, size of the equipment, time frame for usage, and any restrictions on labor hours.

- *Product requirements are the responsibility of the business analyst. Project requirements are the responsibility of the project manager.*

Categorization

Requirement categories are used to help group and structure requirements within the documentation.

The process of categorization helps expose vague, misstated, ambiguous, or otherwise poorly written requirements.

Categorization filters out the bad or poorly written requirements.

Examples of possible filters are:

- **Scope filter.** Determine whether a requirement or information is in scope, out of scope, or unknown.
- **Functional filter.** Once the functional categories have been determined, any defined functional requirement not fitting into one of the categories can be filtered out, revised, or discarded.
- **Prioritization filter.** An arbitrary level of priority (e.g., needs, wants, and desires), is defined and used as a filter to determine which requirements stay or are removed.
- **Testability filter.** All requirements need to be testable, and all requirements should be examined to determine if they are testable. Any requirements that are not testable are filtered out and need to be revised.



Guidelines for Writing Requirements

Information needs to be transcribed into high-quality, well-formatted requirements. Requirements that are well written are of higher value to the solution developer and overall project team, because these will be clear, concise, and reduce conflict and confusion on what needs to be delivered.

A well-formatted requirement consists of the following elements:

- Condition,
- Subject,
- Imperative,
- Active verb,
- Object,
- Business rule (optional), and
- Outcome (optional).

Example—A well-formed detail level requirement might be as follows: When the new account button is pressed (condition), the system (subject) will (imperative) display (active verb) the new account entry screen (object) allowing the creation of a new account (outcome).

Guidelines for Writing Requirements

The following characteristics serve as a checklist when reviewing requirements to ensure they are of high quality.

1. Unambiguous
2. Precise

Ambiguous Requirements	Unambiguous Requirements
The system shall check the name field to be only alphabetic and the address field to be either alphabetic or numeric but containing only addresses in the U.S. or Canada, and the quantity field to be only numeric	3.4.1 The system shall validate that 3.4.1.1 The name field is alphabetic 3.4.1.2 The address field is either alphabetic or numeric 3.4.1.3 All addresses are in the U.S. or Canada only 3.4.1.4 The quantity field is numeric only
The system provides identification of the employee when passing through the reader	3.9.12 When the employee passes through the reader, the system displays the photograph of the employee on the monitor
Imprecise	Precise
9.2.1 When the department code entered does not match the department code on file, the system will display an error message.	9.2.1 When the department code entered does not match the department code on file, the system will display "invalid department code."

Guidelines for Writing Requirements

The following characteristics serve as a checklist when reviewing requirements to ensure they are of high quality.

1. Unambiguous 2. Precise 3. Consistent 4. Correct 5. Complete 6. Measurable 7. Feasible 8. Traceable 9. Testable

Ambiguous Requirements	Unambiguous Requirements
The system shall check the name field to be only alphabetic and the address field to be either alphabetic or numeric but containing only addresses in the U.S. or Canada, and the quantity field to be only numeric	3.4.1 The system shall validate that 3.4.1.1 The name field is alphabetic 3.4.1.2 The address field is either alphabetic or numeric 3.4.1.3 All addresses are in the U.S. or Canada only 3.4.1.4 The quantity field is numeric only
The system provides identification of the employee when passing through the reader	3.9.12 When the employee passes through the reader, the system displays the photograph of the employee on the monitor
Imprecise	Precise
9.2.1 When the department code entered does not match the department code on file, the system will display an error message.	9.2.1 When the department code entered does not match the department code on file, the system will display "invalid department code."

Guidelines for Writing Requirements

Inconsistent	Consistent
17.1.4 The security system will	17.1.4 The security system will
22.4.9 The new security system will	22.4.9 The security system will
33.9.11 The secure card system will	33.9.11 The security system will
34.12.12 The R/F security system will	34.12.12 The security system will
Incorrect	Correct
7.0 Security	7.0 Security
7.7.1 The password will be at least 8 characters in length	7.7.1 The password will be 8 characters in length
11.0 User interface	11.0 User interface
11.9.13 The password will be no less than 8 characters in length	11.9.13 The password length is defined in section 7.7.1

Guidelines for Writing Requirements

Incomplete	Complete
The card reader shall be of the same dimensions as indicated by the card size and consistent with industry standards	54.1 Card reader dimensions TBD by April 3 (by John Doe, Security Architect)
Terminate a session after the number of incorrect passwords exceeds the maximum allowed	24.2.2 Terminate a session after the 3 incorrect passwords have been entered
Not Measurable	Measurable
There will be no more than 6 training classes per employee	Each employee will have not less than 2 and no more than 6 training classes residing on their professional development profile
The new production line shall be efficient	The new production line shall produce an average of 5000 bottle caps per day

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:

Document the solution requirements

- Why Documentation is Important
- Business Requirements Document
- Prioritizing Requirements (Prioritization Schemes)
- Technical Requirements Specification
 - Documenting with Use Cases
 - Documenting with User Stories
 - Backlog Items

Prioritization Techniques

MoSCoW. MoSCow establishes a set of prioritization rules which are:

- Must have (fundamental to project success),
- Should have (important, but the project success does not rely on them),
- Could have (can easily be left out without impacting the project), and
- Won't have (not delivered this time around).

Multivoting

Time-boxing

Weighted ranking

REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



Validate Requirements

- The Concept of Continual Confirmation
- Requirements Walkthrough

Verify Requirements

- Peer Review
- Inspection

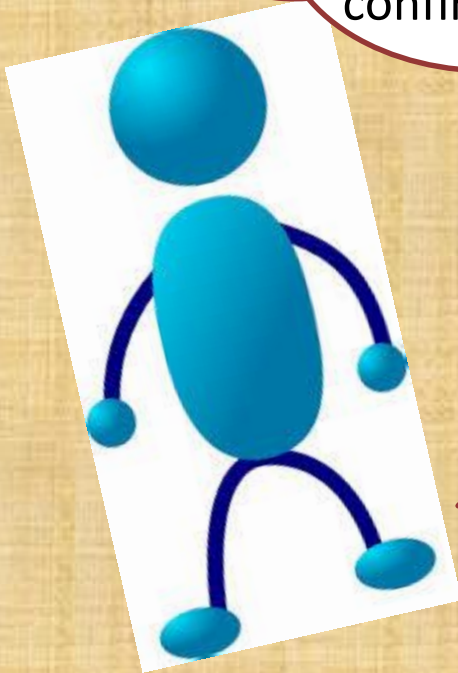
REQUIREMENTS ELICITATION AND ANALYSIS

Validate Requirements process ensures all requirements accurately reflect the intent of the stakeholder.

It is usually performed via requirements walkthrough or concept of continual confirmation

Verify Requirements process reviews the requirements for errors and quality criteria

It is usually performed via peer review or inspection.



REQUIREMENTS ELICITATION AND ANALYSIS

Eliciting requirements and analysis involves:



THE END