UNIT-I

USE CASE

MADHESWARI.K
AP/CSE
SSNCE

What is USE CASE?

- Use cases are text stories – widely used to discover and record requirements
- Informally, **use cases** are ***text stories*** of some **actor** using a **system** to meet **goals**.
- Use Case Example :

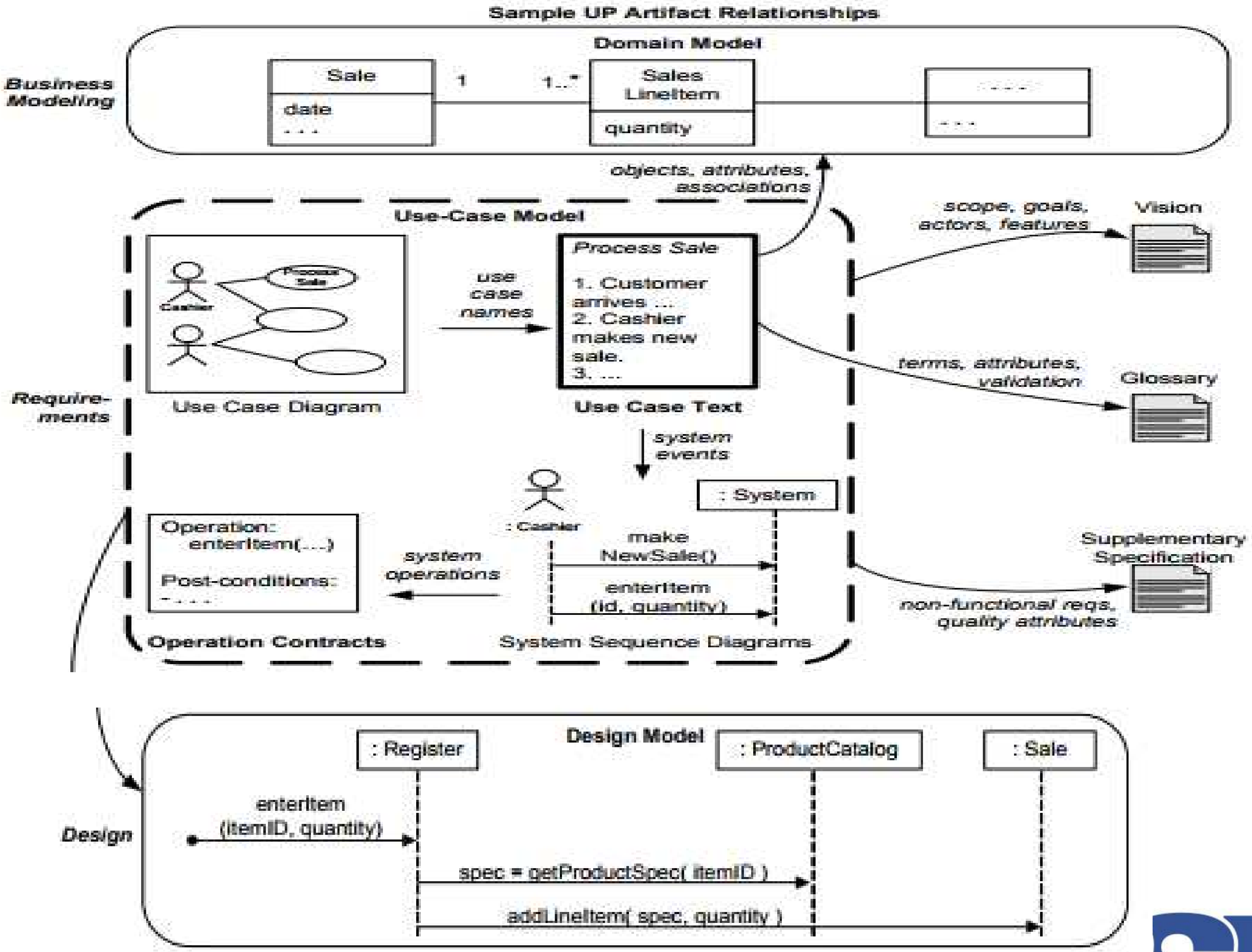
Process Sale: A customer arrives at a checkout with items to purchase. The cashier uses the **POS** system to record each purchased item. The system presents a **running total** and **line-item** details. The customer enters **payment** information, which the system validates and records. The system updates i**nventory**. The customer receives a **receipt** from the system and then leaves with the items.

What is USE CASE?

- Notice that ***use cases are not diagrams, they are text*.** Focusing on secondary-value UML use case diagrams rather than the important use case text is a common mistake for use case novices.

- Use cases often need to be more detailed or structured than this example, but the essence is **discovering and recording functional requirements by writing stories** of using a system to fulfill user goals; that is, *cases of use*.

What is USE CASE?



Definition: What are Actors, Scenarios, and Use Cases?

Actor

- **An actor** is something with behavior, such as a person (identified by role), computer system, or organization; for example, a cashier.

Scenario

- **A scenario** is a specific sequence of actions and interactions between actors and the system; it is also called a use case instance.
- It is one particular story of using a system, or one path through the use case; for example, the scenario of successfully purchasing items with cash, or the scenario of failing to purchase items because of a credit payment denial.

Definition: What are Actors, Scenarios, and Use Cases?

Use case

- Informally then, **a use case** is a collection of related success and failure scenarios that describe an actor using a system to support a goal

Definition of a use case provided by the RUP

- A set of use-case instances, where each instance is a sequence of actions a system performs that yields an observable result of value to a particular actor [RUP].

Use-case Modeling

- **Use-Case Model** is the set of all written use cases; it is a model of the system's functionality and environment.

Use cases are text documents, not diagrams, and use-case modeling is primarily an act of writing text, not drawing diagrams.

Other UP requirements

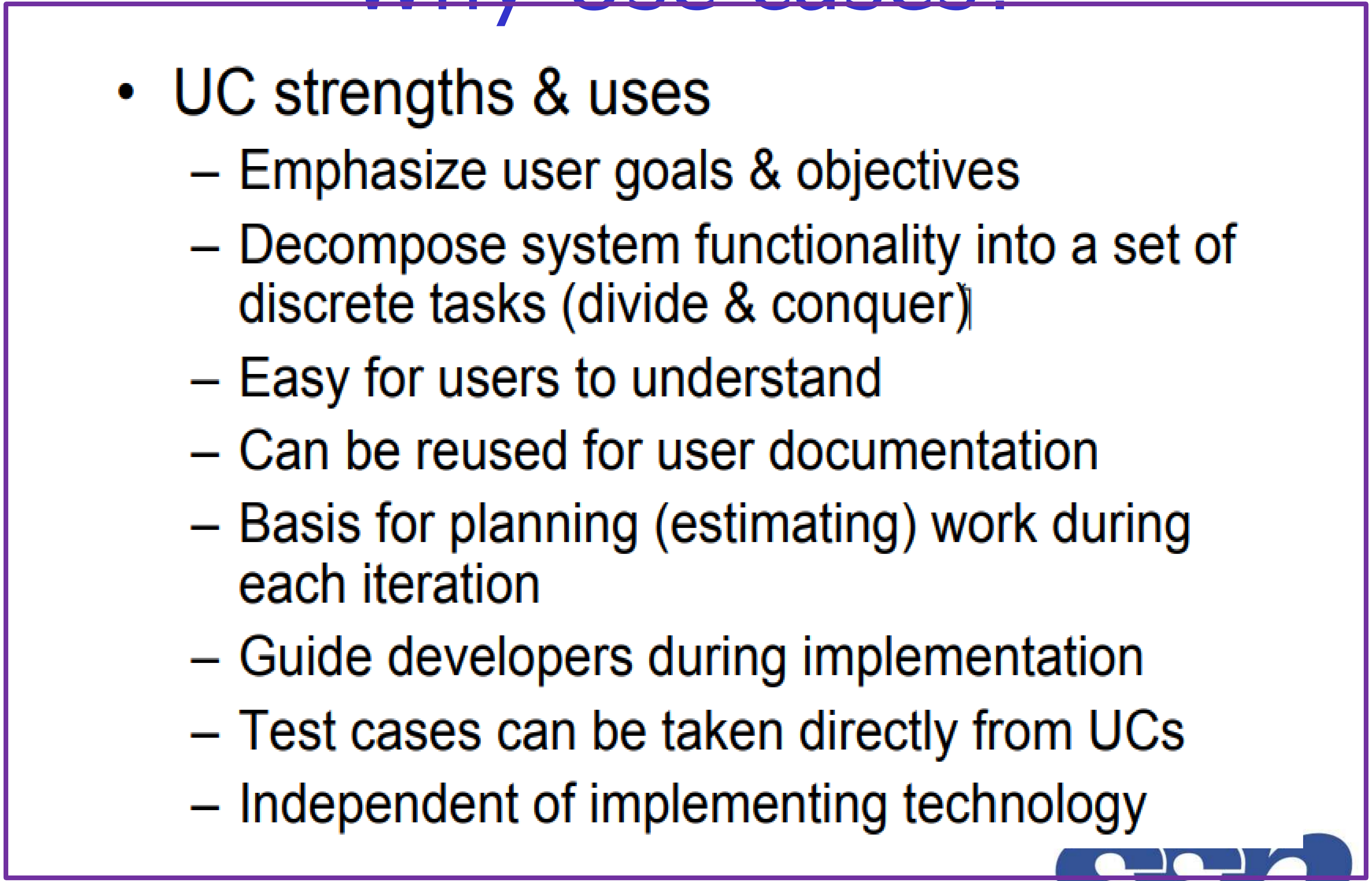
- The Use-Case Model is not the only requirement artifact in the UP.
- There are also the **Supplementary Specification, Glossary, Vision, and Business Rules**.
- These are all useful for requirements analysis, but secondary at this point.

Use-case Modeling

UML use case diagram

- The Use-Case Model may optionally include a **UML use case diagram** to show the **names** of **use cases and actors, and their relationships.**
- This gives a nice **context diagram** of a system and its environment.
- It also provides a quick way to list the use cases by name.
- There is nothing object-oriented about use cases; we're not doing OO analysis when writing them. Use cases are a key requirements input to classic OOA/D.

Why Use cases?



- UC strengths & uses
  - Emphasize user goals & objectives
  - Decompose system functionality into a set of discrete tasks (divide & conquer)
  - Easy for users to understand
  - Can be reused for user documentation
  - Basis for planning (estimating) work during each iteration
  - Guide developers during implementation
  - Test cases can be taken directly from UCs
  - Independent of implementing technology

Three kinds of Actors

- **Actors** are roles played not only by people, but by organizations, software, and machines.
- There are three kinds of external actors in relation to the SuD:

1. Primary actor
2. Supporting actor
3. Offstage actor

Primary actor

- **Primary actor** has user goals fulfilled through using services of the SuD. For example, the cashier.
- Why identify? To find user goals, which drive the use cases**.**

Three kinds of Actors

Supporting actor

- **Supporting actor** provides a service (for example, information) to the SuD. The automated payment authorization service is an example. Often a computer system, but could be an organization or person.
- Why identify? To clarify external interfaces and protocols.

Offstage actor

- **Offstage actor** has an interest in the behavior of the use case, but is not primary or supporting; for example, a government tax agency.
- Why identify? To ensure that *all* necessary interests are identified and satisfied.

Three common use case format

Use cases can be written in different formats and levels of formality

1. Brief
2. Casual
3. Fully dressed

Brief

- One paragraph summary, usually of the main success scenario
- When? During early requirement analysis, to get a quick sense of subject and scope
- May take only few minutes to create.

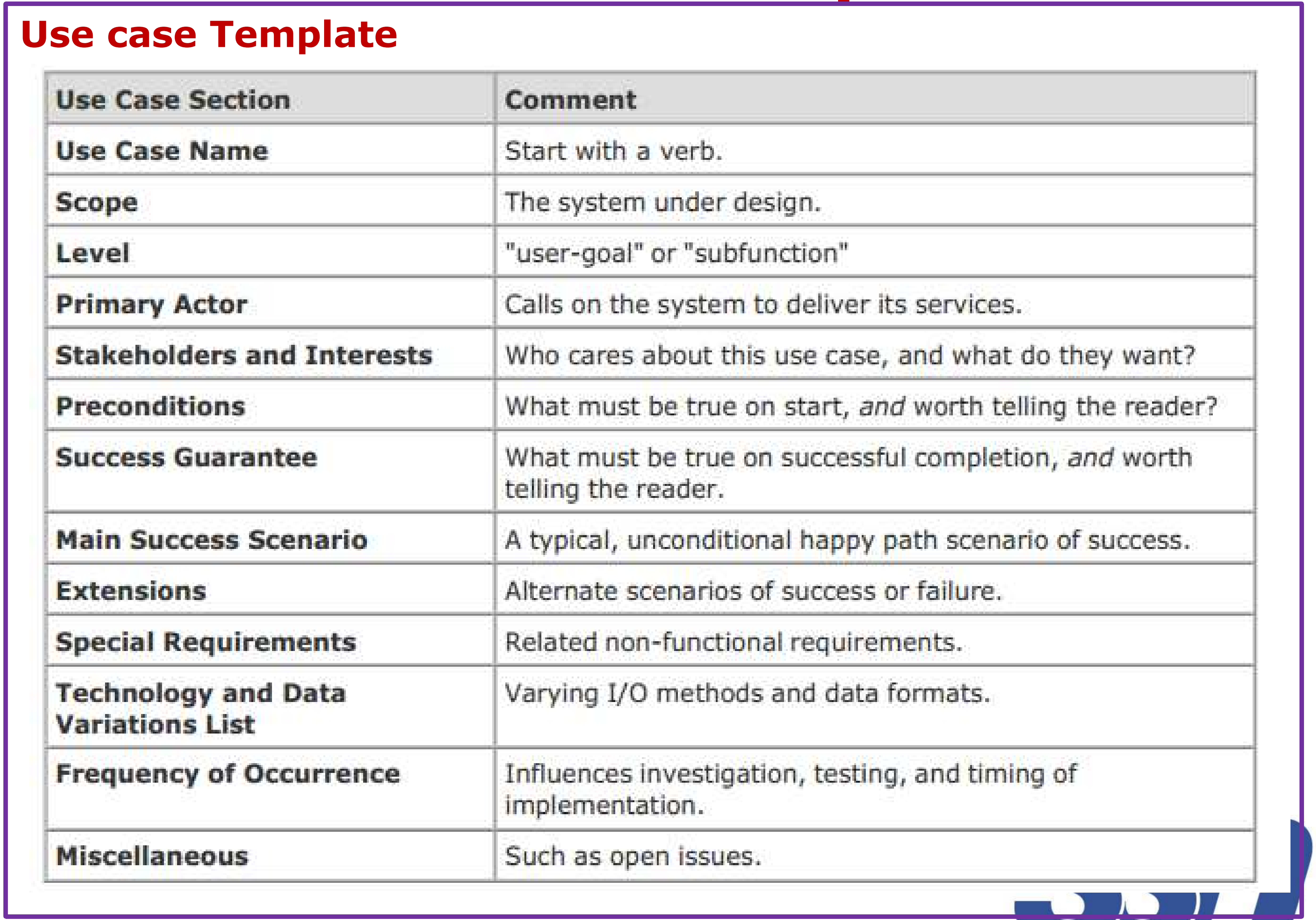
Casual

- Informal paragraph format. Multiple paragraphs that cover various scenarios.

Fully dressed

- All steps and variations are written in detail, and there are supporting sub sections, such as pre conditions and post conditions.

Use case Template



Use case Template

Use Case Section	Comment
Use Case Name	Start with a verb.
Scope	The system under design.
Level	"user-goal" or "subfunction"
Primary Actor	Calls on the system to deliver its services.
Stakeholders and Interests	Who cares about this use case, and what do they want?
Preconditions	What must be true on start, *and* worth telling the reader?
Success Guarantee	What must be true on successful completion, *and* worth telling the reader.
Main Success Scenario	A typical, unconditional happy path scenario of success.
Extensions	Alternate scenarios of success or failure.
Special Requirements	Related non-functional requirements.
Technology and Data Variations List	Varying I/O methods and data formats.
Frequency of Occurrence	Influences investigation, testing, and timing of implementation.
Miscellaneous	Such as open issues.

Example: use case

Following is an example of informal description or text story for withdrawal of money from ATM

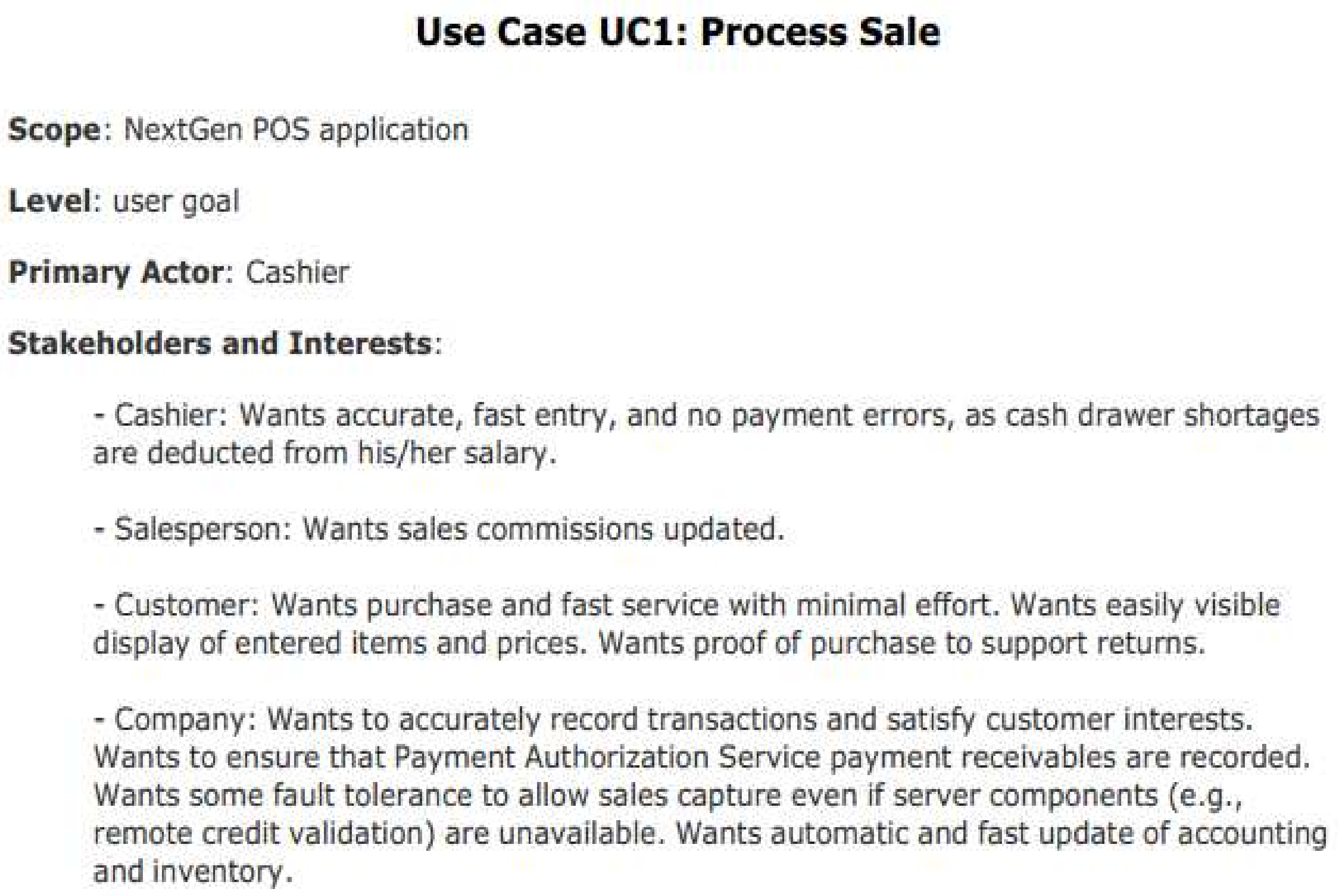
"Customer wants to operate the ATM machine for performing some transactions. For instance customer wants to withdraw the money. He enters the ATM card and then types in the PIN. The system validates the customer. If the customer is the valid customer then the customer is allowed to do further transactions. Otherwise the card will be rejected. The valid customer enters the amount to be withdrawn. It is then checked if withdrawal amount < balance amount. If it is so, then the machine dispense the desired amount. If there is no further transactions then the card is ejected. Customer collects the cash, statement and card."

Example: use case

Use case	ATM System
Primary Actor	Customer
Goal in the context	To monitor all the functionalities required to establish the connection
Preconditions	ATM system has to be programmed and as to recognize and validate the PIN
Trigger	On completion of every activity a beep has to be generated by the system
Scenario	1. Customer observes the control panel 2. Customer enters the ATM card 3. Customer enters the PIN 4. Customer selects the operation(withdrawal, deposit, inquiry) 5. Customer collects the cash, statement, card etc.
Exceptions	1. The control panel is not ready 2. PIN is incorrect 3. Card is not recognized 4. Insufficient balance 5. Limit for the transaction exceeds 6. Total number of allowed transactions per day
Priority	Essential and must be implemented in banking system
Secondary actor	Administrator
Open issues	1. For how many time the PIN is allowed to enter on incorrect supplement

Example 2: process sale, Fully dressed Style

Use case Example



Use Case UC1: Process Sale

Scope: NextGen POS application

Level: user goal

Primary Actor: Cashier

Stakeholders and Interests:

- Cashier: Wants accurate, fast entry, and no payment errors, as cash drawer shortages are deducted from his/her salary.

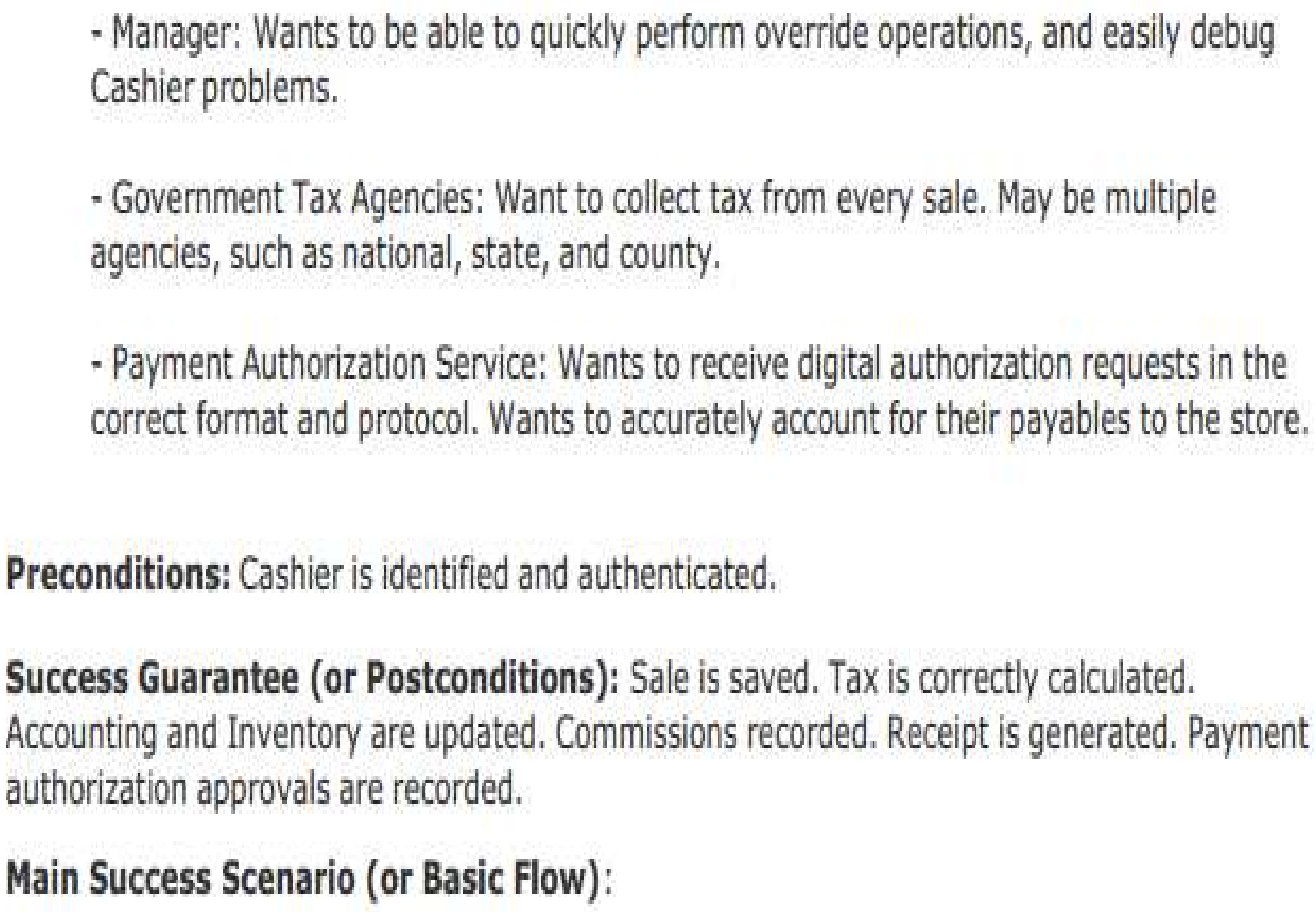
- Salesperson: Wants sales commissions updated.

- Customer: Wants purchase and fast service with minimal effort. Wants easily visible display of entered items and prices. Wants proof of purchase to support returns.

- Company: Wants to accurately record transactions and satisfy customer interests. Wants to ensure that Payment Authorization Service payment receivables are recorded. Wants some fault tolerance to allow sales capture even if server components (e.g., remote credit validation) are unavailable. Wants automatic and fast update of accounting and inventory.

Example: process sale, Fully dressed Style

Use case Example



- Manager: Wants to be able to quickly perform override operations, and easily debug Cashier problems.

- Government Tax Agencies: Want to collect tax from every sale. May be multiple agencies, such as national, state, and county.

- Payment Authorization Service: Wants to receive digital authorization requests in the correct format and protocol. Wants to accurately account for their payables to the store.

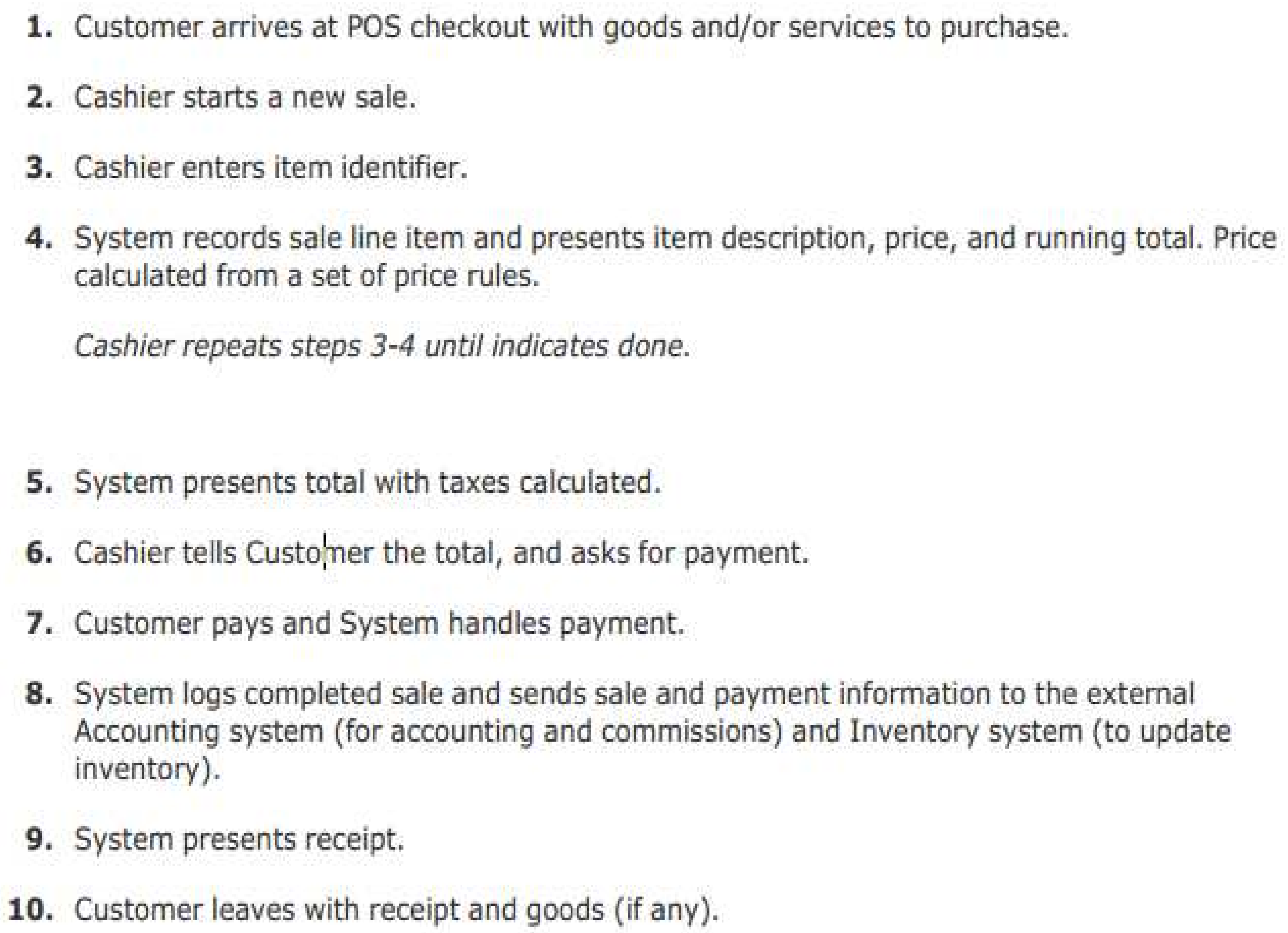
Preconditions: Cashier is identified and authenticated.

Success Guarantee (or Postconditions): Sale is saved. Tax is correctly calculated. Accounting and Inventory are updated. Commissions recorded. Receipt is generated. Payment authorization approvals are recorded.

Main Success Scenario (or Basic Flow):

Example: process sale, Fully dressed Style

Use case Example



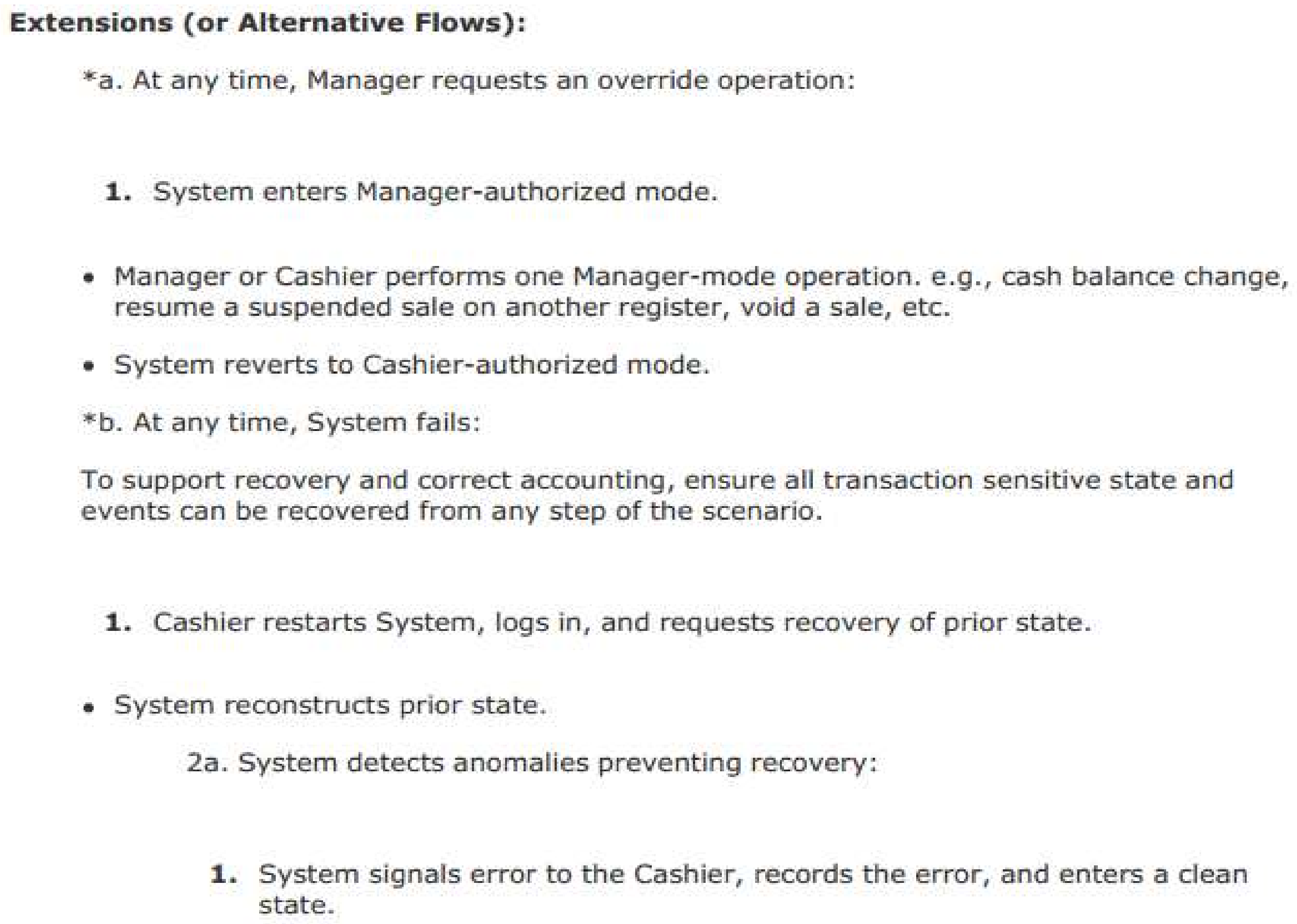
1. Customer arrives at POS checkout with goods and/or services to purchase.
2. Cashier starts a new sale.
3. Cashier enters item identifier.
4. System records sale line item and presents item description, price, and running total. Price calculated from a set of price rules.

   Cashier repeats steps 3-4 until indicates done.

5. System presents total with taxes calculated.
6. Cashier tells Customer the total, and asks for payment.
7. Customer pays and System handles payment.
8. System logs completed sale and sends sale and payment information to the external Accounting system (for accounting and commissions) and Inventory system (to update inventory).
9. System presents receipt.
10. Customer leaves with receipt and goods (if any).

Example: process sale, Fully dressed Style

Use case Example



Extensions (or Alternative Flows):

*a. At any time, Manager requests an override operation:

1. System enters Manager-authorized mode.

- Manager or Cashier performs one Manager-mode operation. e.g., cash balance change, resume a suspended sale on another register, void a sale, etc.
- System reverts to Cashier-authorized mode.

*b. At any time, System fails:

To support recovery and correct accounting, ensure all transaction sensitive state and events can be recovered from any step of the scenario.

1. Cashier restarts System, logs in, and requests recovery of prior state.

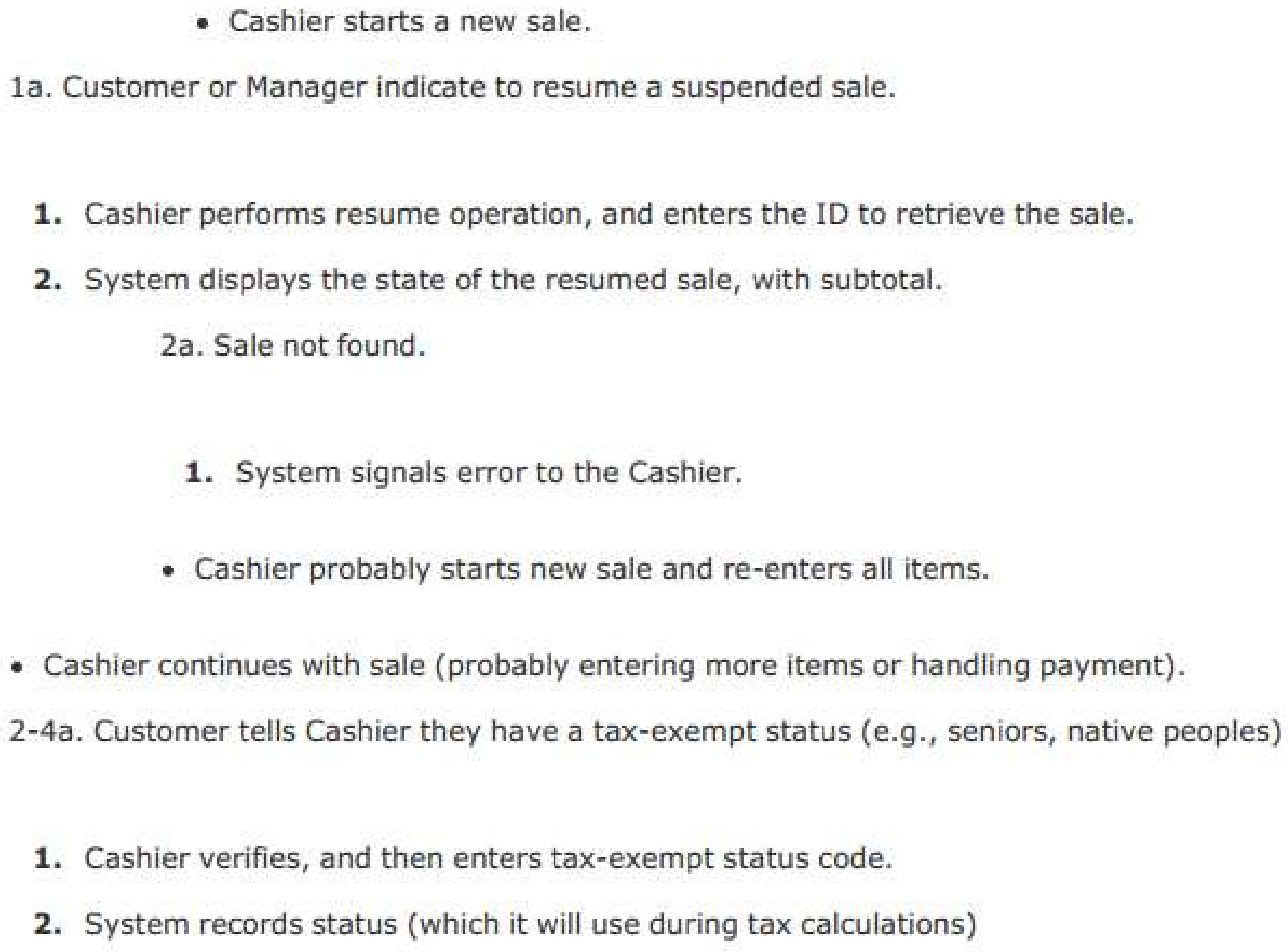
- System reconstructs prior state.

2a. System detects anomalies preventing recovery:

1. System signals error to the Cashier, records the error, and enters a clean state.

Example: process sale, Fully dressed Style

Use case Example



- Cashier starts a new sale.

1a. Customer or Manager indicate to resume a suspended sale.

1. Cashier performs resume operation, and enters the ID to retrieve the sale.
2. System displays the state of the resumed sale, with subtotal.

    2a. Sale not found.

    1. System signals error to the Cashier.

    - Cashier probably starts new sale and re-enters all items.

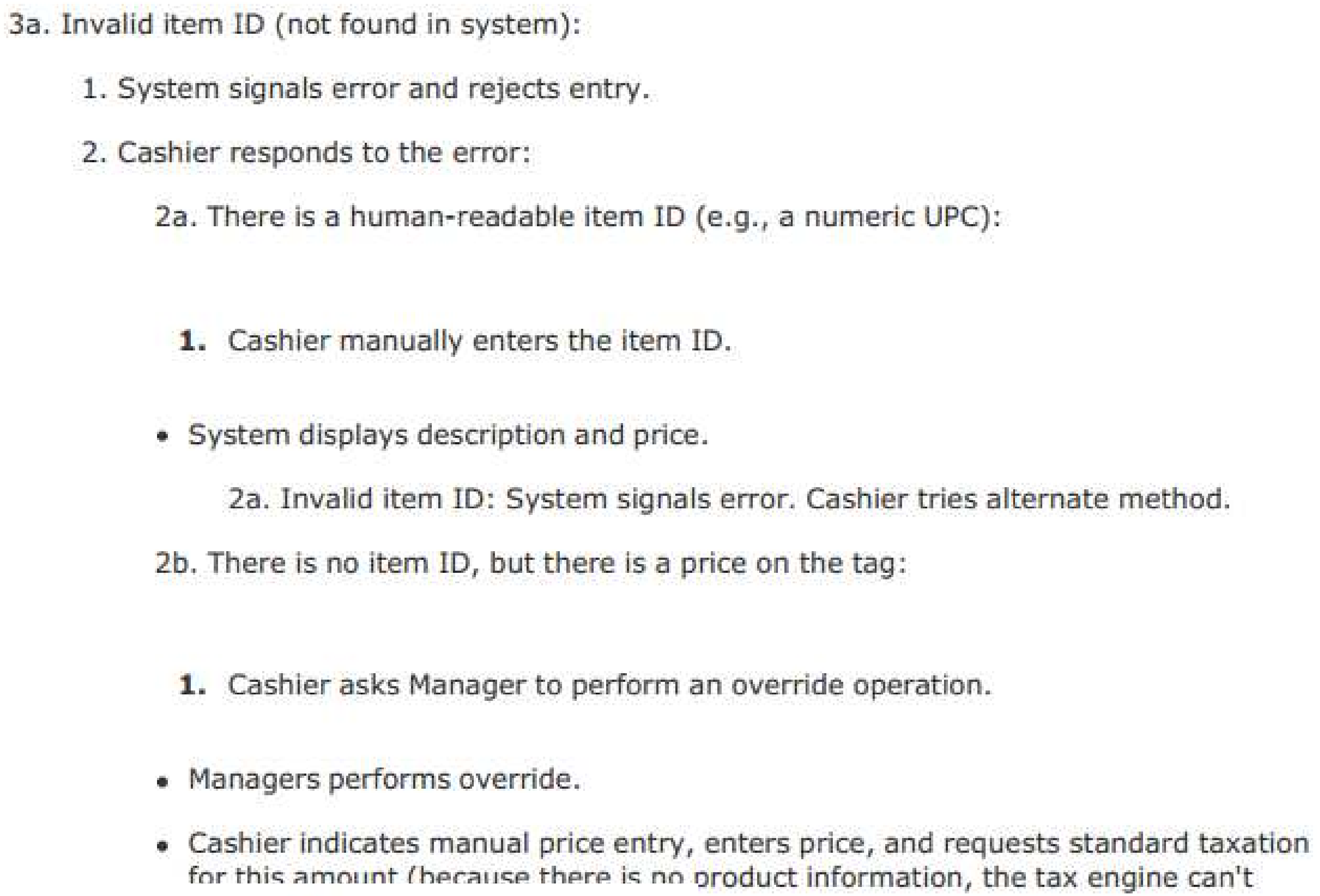
- Cashier continues with sale (probably entering more items or handling payment).

2-4a. Customer tells Cashier they have a tax-exempt status (e.g., seniors, native peoples)

1. Cashier verifies, and then enters tax-exempt status code.
2. System records status (which it will use during tax calculations)

Example: process sale, Fully dressed Style

Use case Example



3a. Invalid item ID (not found in system):

1. System signals error and rejects entry.
2. Cashier responds to the error:

   2a. There is a human-readable item ID (e.g., a numeric UPC):

   1. Cashier manually enters the item ID.

   - System displays description and price.

     2a. Invalid item ID: System signals error. Cashier tries alternate method.

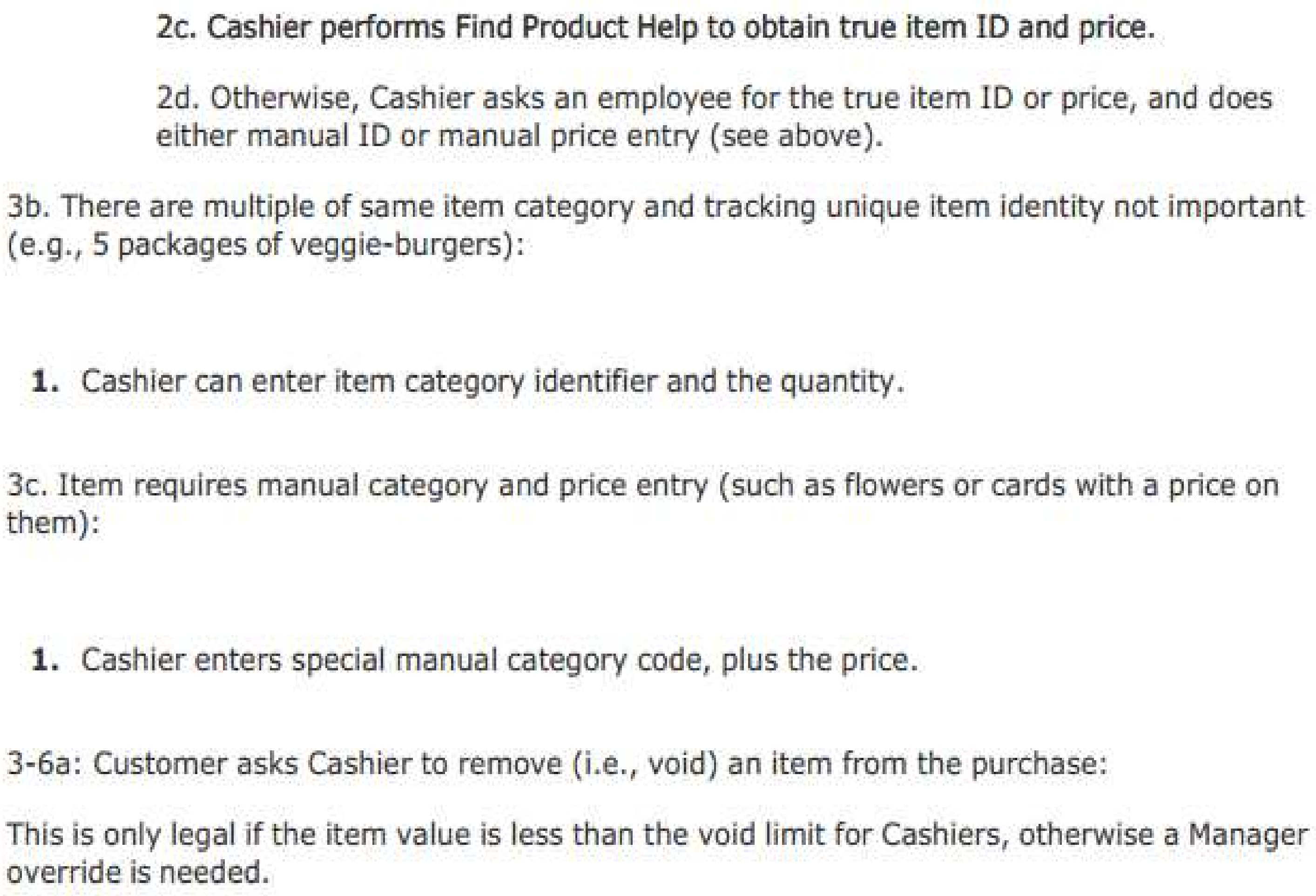
   2b. There is no item ID, but there is a price on the tag:

   1. Cashier asks Manager to perform an override operation.

   - Managers performs override.
   - Cashier indicates manual price entry, enters price, and requests standard taxation for this amount (because there is no product information, the tax engine can't otherwise deduce how to tax it)

Example: process sale, Fully dressed Style

Use case Example



2c. Cashier performs Find Product Help to obtain true item ID and price.

2d. Otherwise, Cashier asks an employee for the true item ID or price, and does either manual ID or manual price entry (see above).

3b. There are multiple of same item category and tracking unique item identity not important (e.g., 5 packages of veggie-burgers):

1. Cashier can enter item category identifier and the quantity.

3c. Item requires manual category and price entry (such as flowers or cards with a price on them):

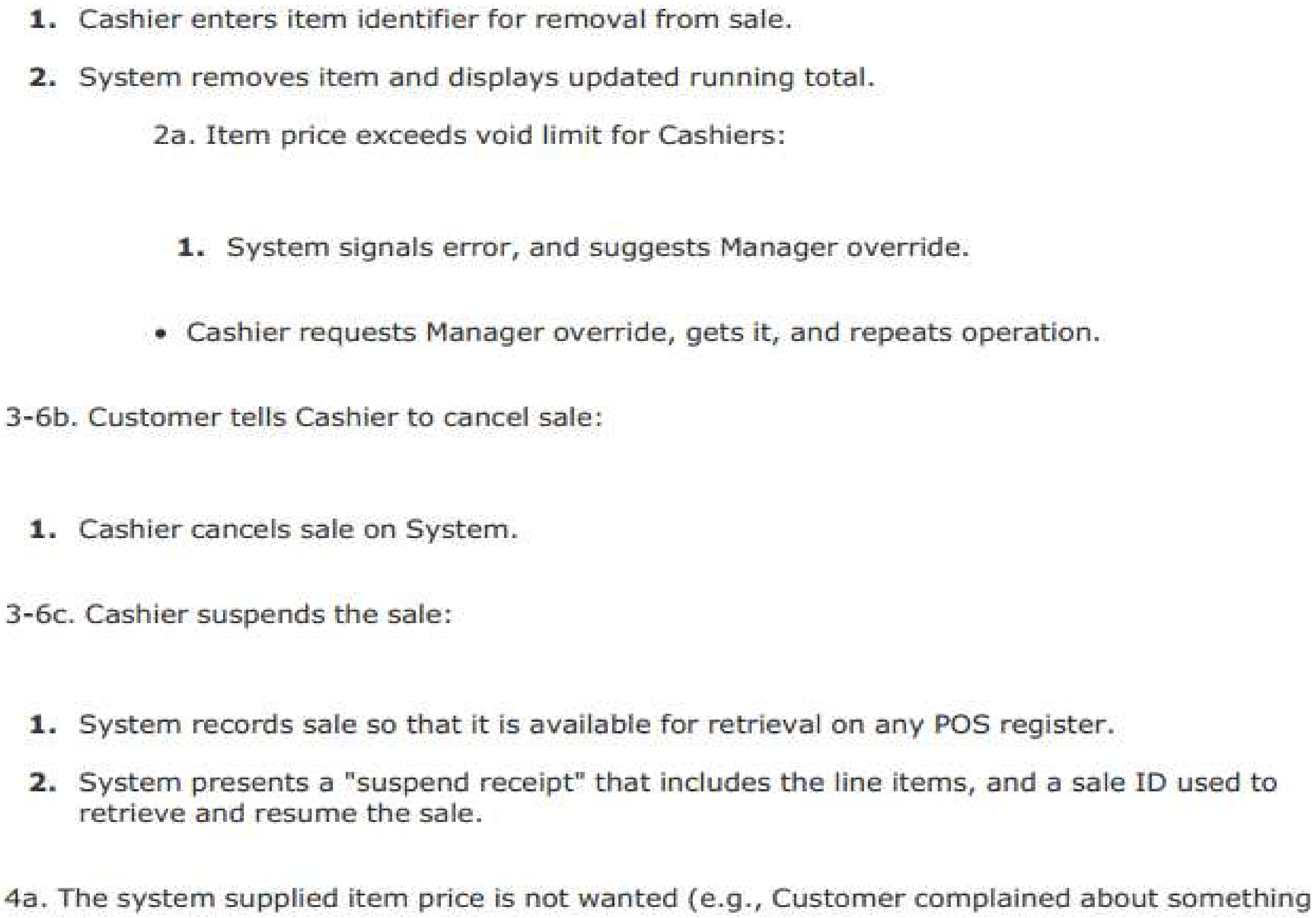
1. Cashier enters special manual category code, plus the price.

3-6a: Customer asks Cashier to remove (i.e., void) an item from the purchase:

This is only legal if the item value is less than the void limit for Cashiers, otherwise a Manager override is needed.

Example: process sale, Fully dressed Style

Use case Example



1. Cashier enters item identifier for removal from sale.
2. System removes item and displays updated running total.

2a. Item price exceeds void limit for Cashiers:

1. System signals error, and suggests Manager override.

- Cashier requests Manager override, gets it, and repeats operation.

3-6b. Customer tells Cashier to cancel sale:

1. Cashier cancels sale on System.

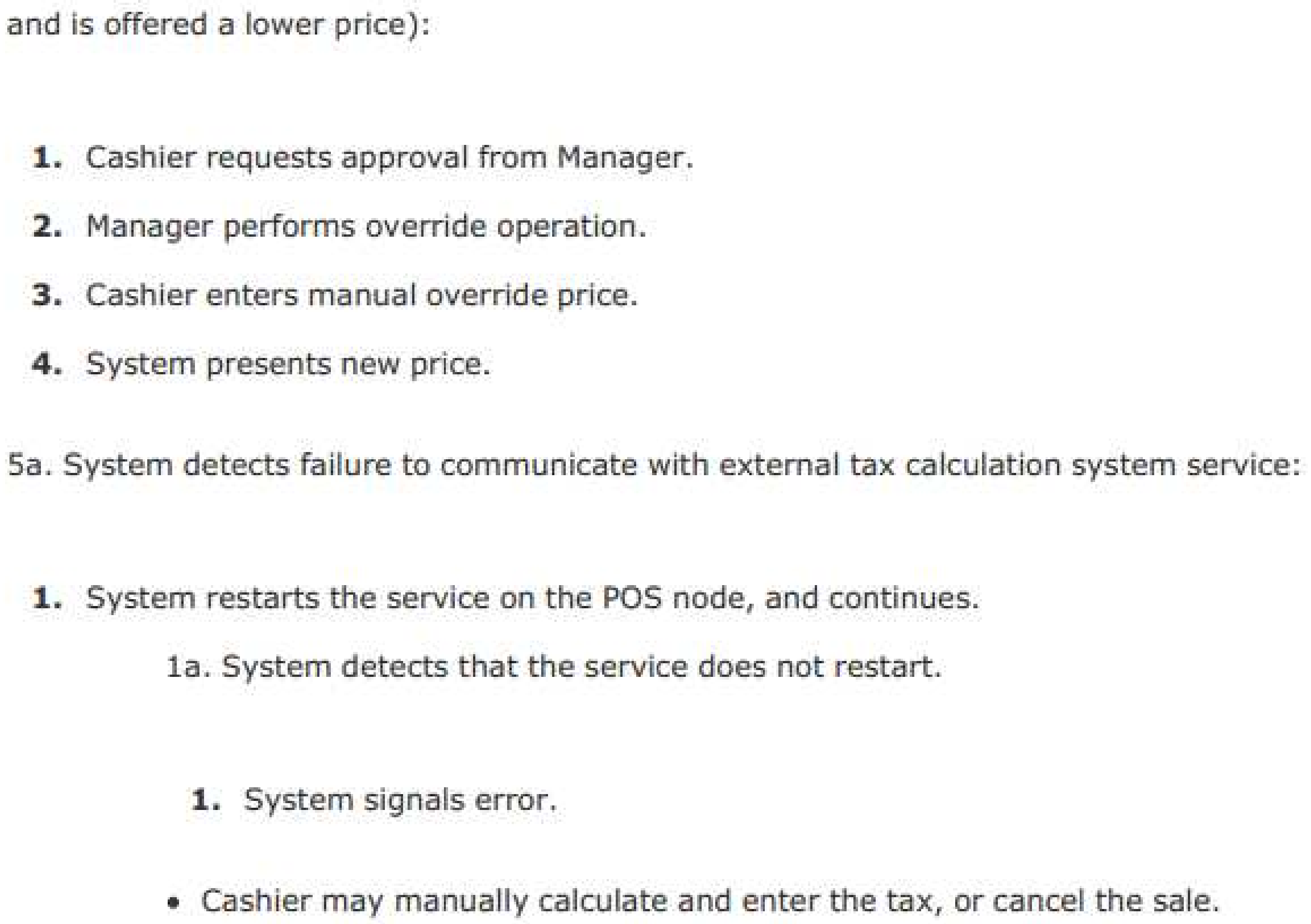
3-6c. Cashier suspends the sale:

1. System records sale so that it is available for retrieval on any POS register.
2. System presents a "suspend receipt" that includes the line items, and a sale ID used to retrieve and resume the sale.

4a. The system supplied item price is not wanted (e.g., Customer complained about something

Example: process sale, Fully dressed Style

Use case Example



and is offered a lower price):

1. Cashier requests approval from Manager.
2. Manager performs override operation.
3. Cashier enters manual override price.
4. System presents new price.

5a. System detects failure to communicate with external tax calculation system service:

1. System restarts the service on the POS node, and continues.

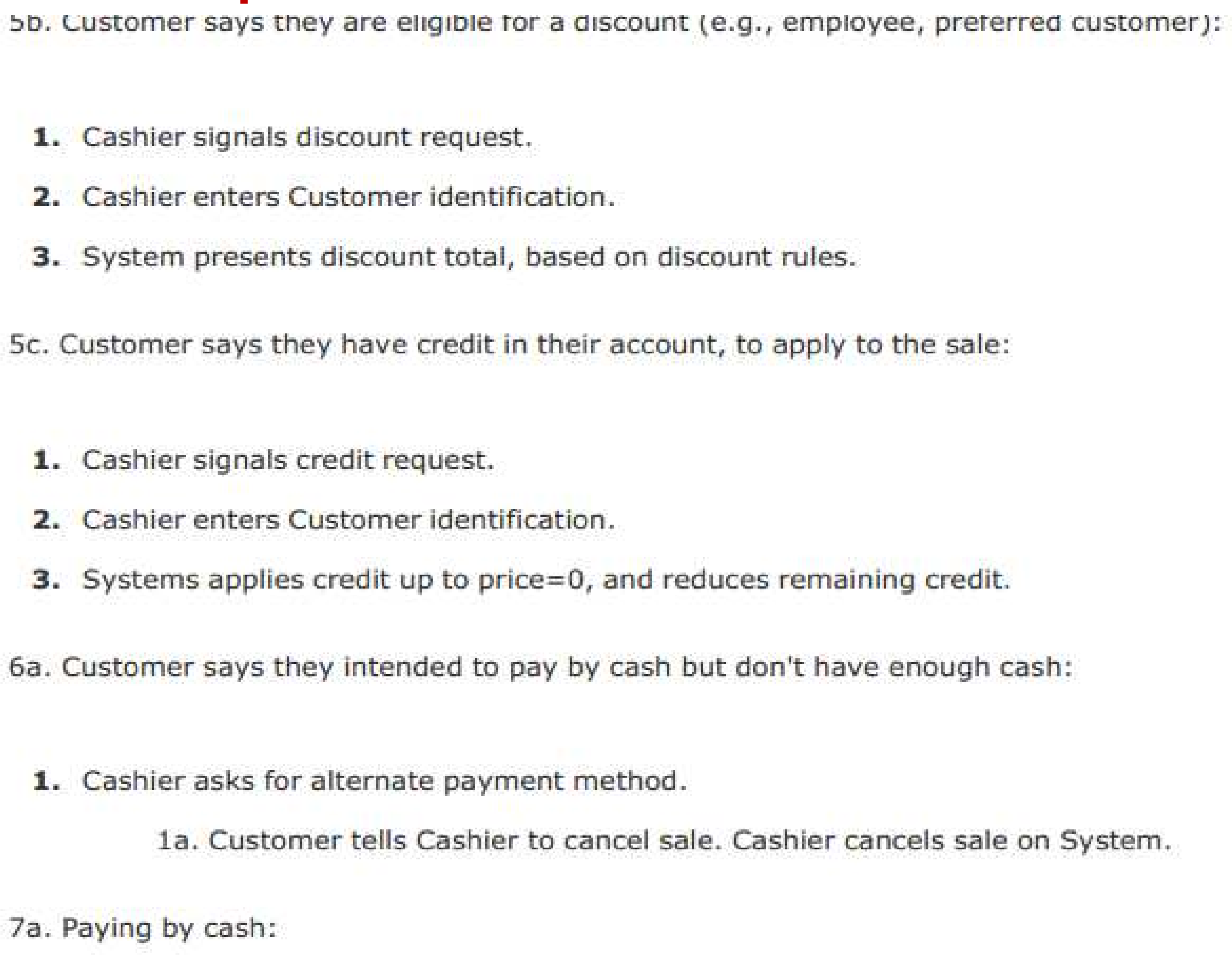
    1a. System detects that the service does not restart.

    1. System signals error.

    - Cashier may manually calculate and enter the tax, or cancel the sale.

Example: process sale, Fully dressed Style

Use case Example



5b. Customer says they are eligible for a discount (e.g., employee, preferred customer):

1. Cashier signals discount request.
2. Cashier enters Customer identification.
3. System presents discount total, based on discount rules.

5c. Customer says they have credit in their account, to apply to the sale:

1. Cashier signals credit request.
2. Cashier enters Customer identification.
3. Systems applies credit up to price=0, and reduces remaining credit.

6a. Customer says they intended to pay by cash but don't have enough cash:

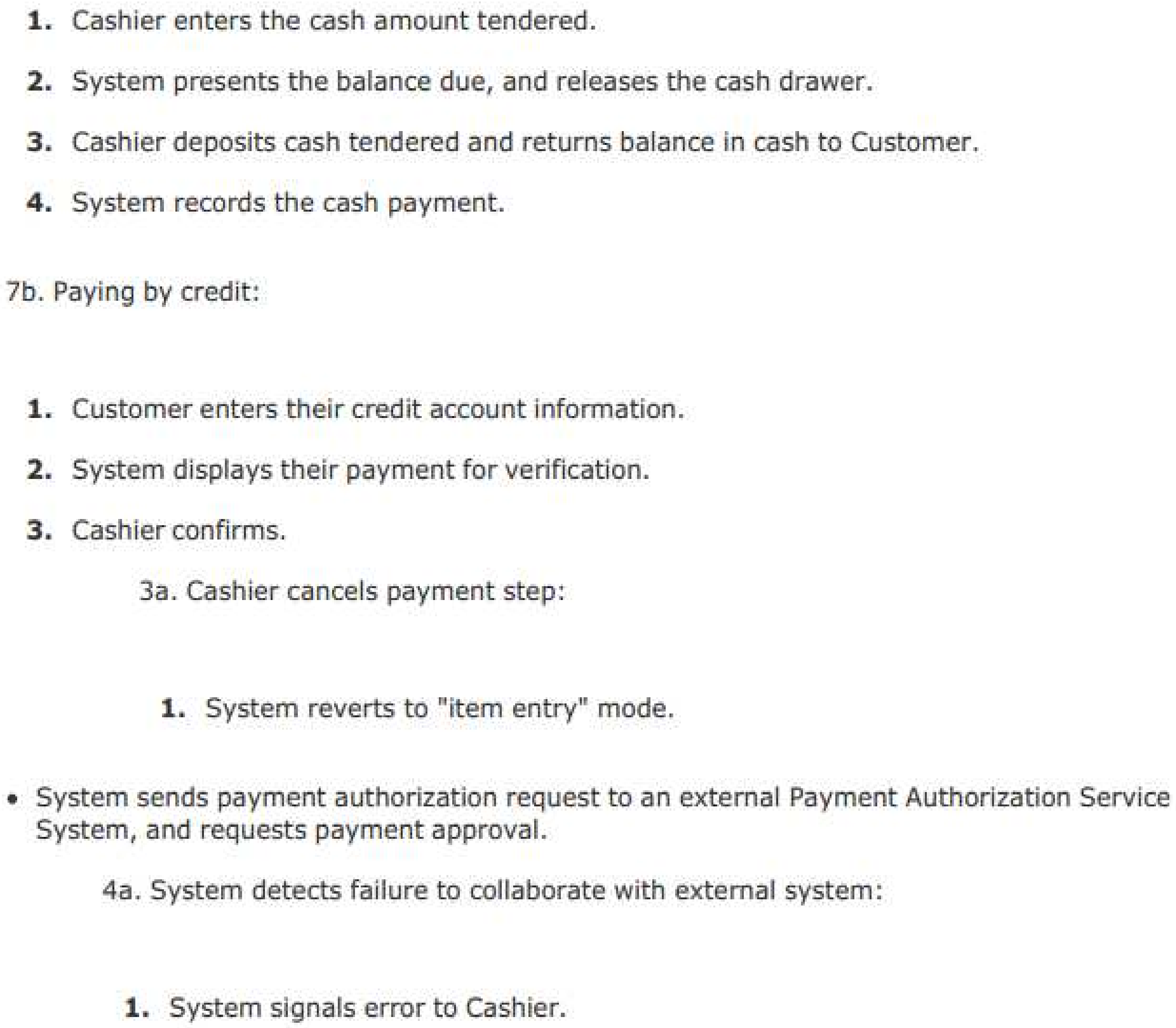
1. Cashier asks for alternate payment method.

    1a. Customer tells Cashier to cancel sale. Cashier cancels sale on System.

7a. Paying by cash:

Example: process sale, Fully dressed Style

Use case Example



1. Cashier enters the cash amount tendered.
2. System presents the balance due, and releases the cash drawer.
3. Cashier deposits cash tendered and returns balance in cash to Customer.
4. System records the cash payment.

7b. Paying by credit:

1. Customer enters their credit account information.
2. System displays their payment for verification.
3. Cashier confirms.

   3a. Cashier cancels payment step:

   1. System reverts to "item entry" mode.

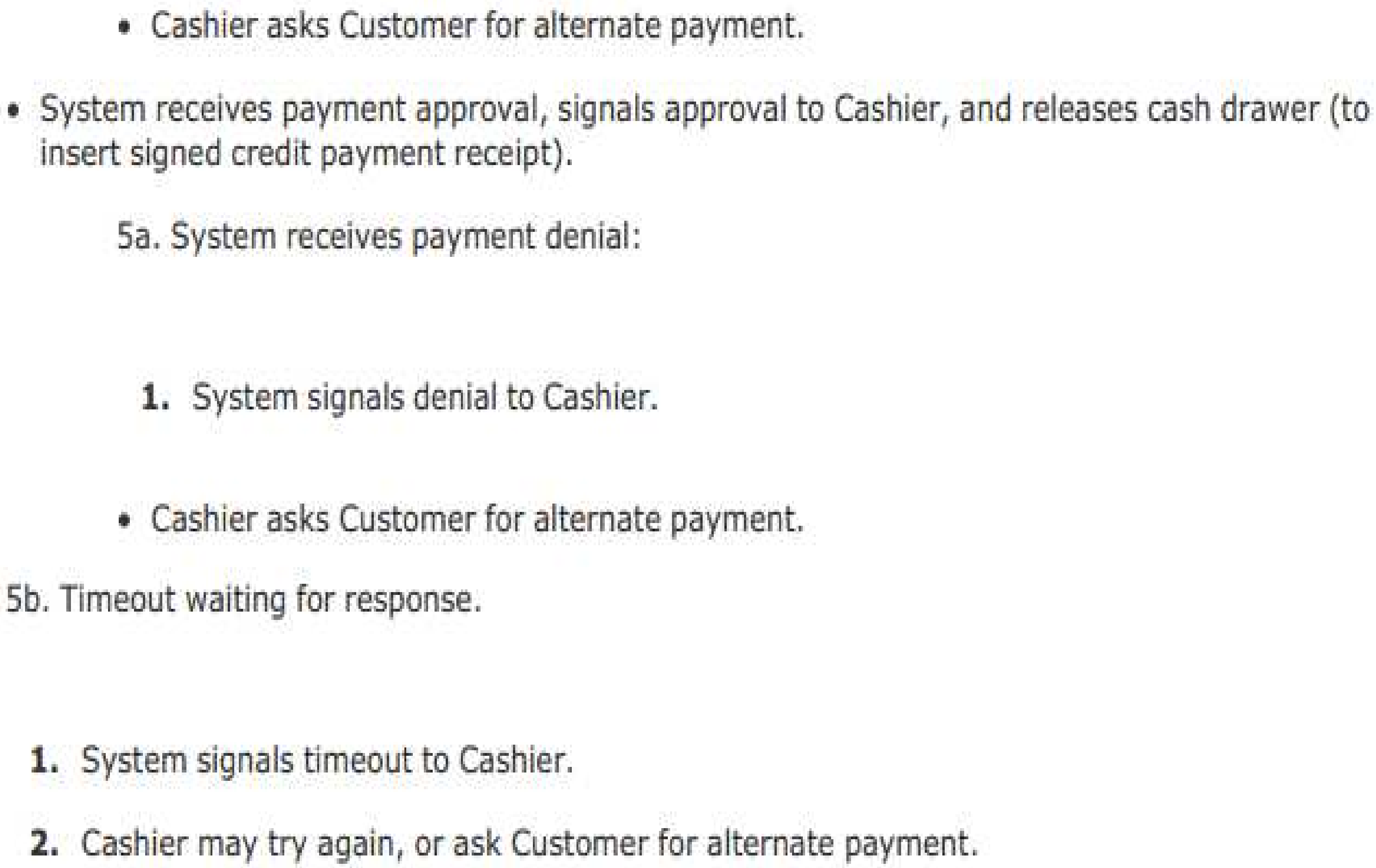
- System sends payment authorization request to an external Payment Authorization Service System, and requests payment approval.

  4a. System detects failure to collaborate with external system:

  1. System signals error to Cashier.

Example: process sale, Fully dressed Style

Use case Example



- Cashier asks Customer for alternate payment.

- System receives payment approval, signals approval to Cashier, and releases cash drawer (to insert signed credit payment receipt).

5a. System receives payment denial:

1. System signals denial to Cashier.

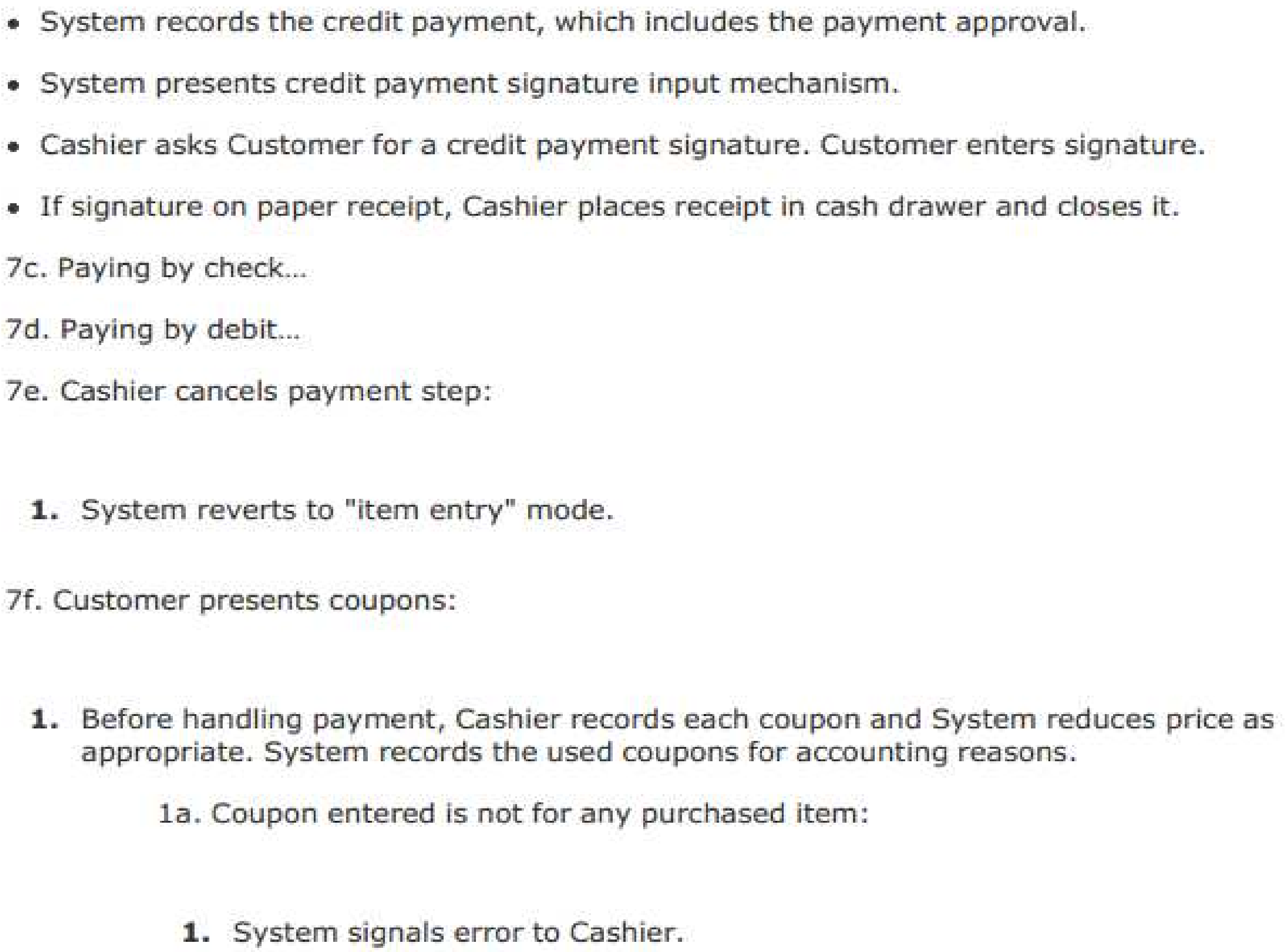
- Cashier asks Customer for alternate payment.

5b. Timeout waiting for response.

1. System signals timeout to Cashier.
2. Cashier may try again, or ask Customer for alternate payment.

Example: process sale, Fully dressed Style

Use case Example



- System records the credit payment, which includes the payment approval.
- System presents credit payment signature input mechanism.
- Cashier asks Customer for a credit payment signature. Customer enters signature.
- If signature on paper receipt, Cashier places receipt in cash drawer and closes it.

7c. Paying by check...

7d. Paying by debit...

7e. Cashier cancels payment step:

1. System reverts to "item entry" mode.

7f. Customer presents coupons:

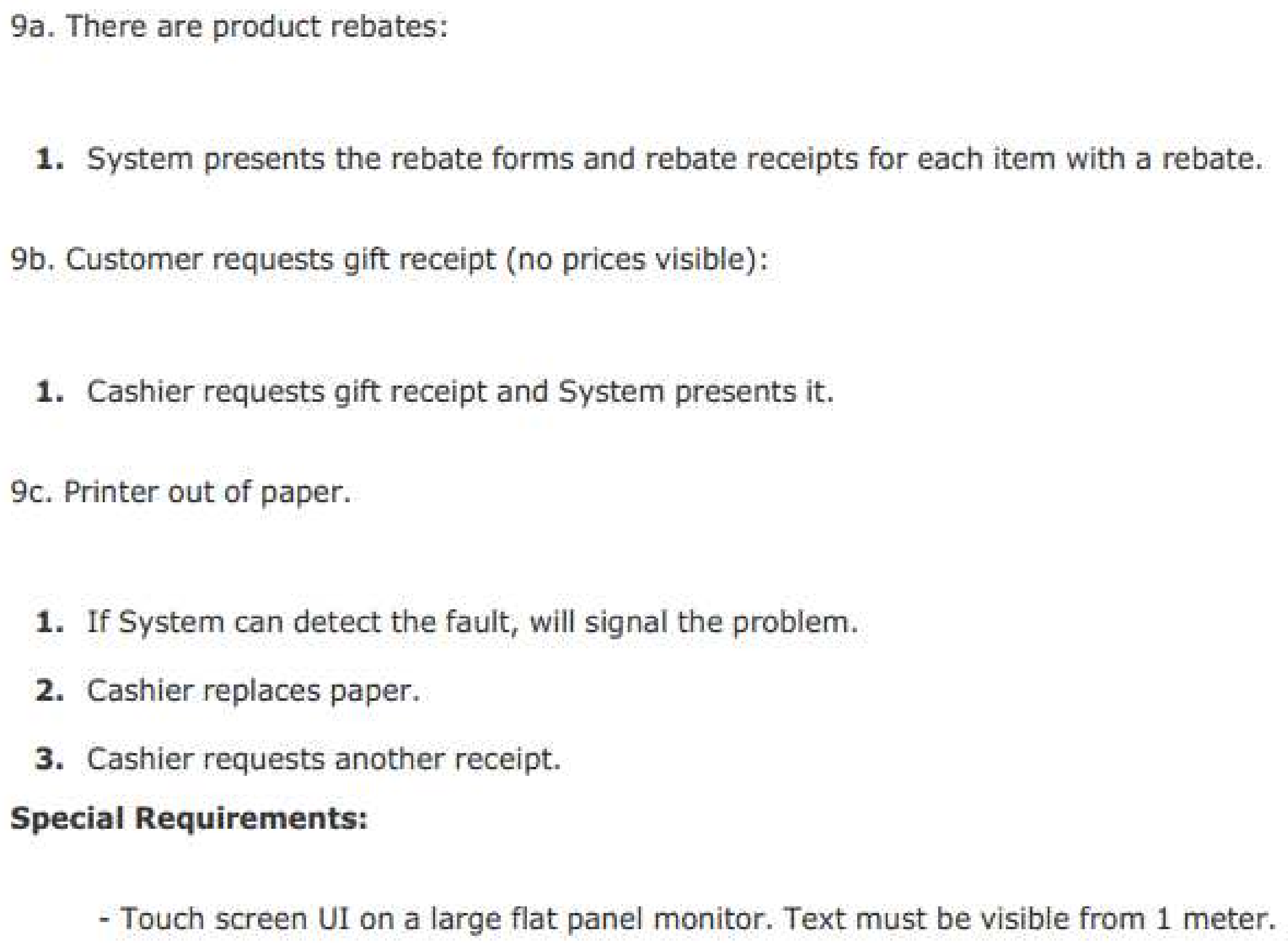
1. Before handling payment, Cashier records each coupon and System reduces price as appropriate. System records the used coupons for accounting reasons.

    1a. Coupon entered is not for any purchased item:

    1. System signals error to Cashier.

Example: process sale, Fully dressed Style

Use case Example



9a. There are product rebates:

1. System presents the rebate forms and rebate receipts for each item with a rebate.

9b. Customer requests gift receipt (no prices visible):

1. Cashier requests gift receipt and System presents it.

9c. Printer out of paper.

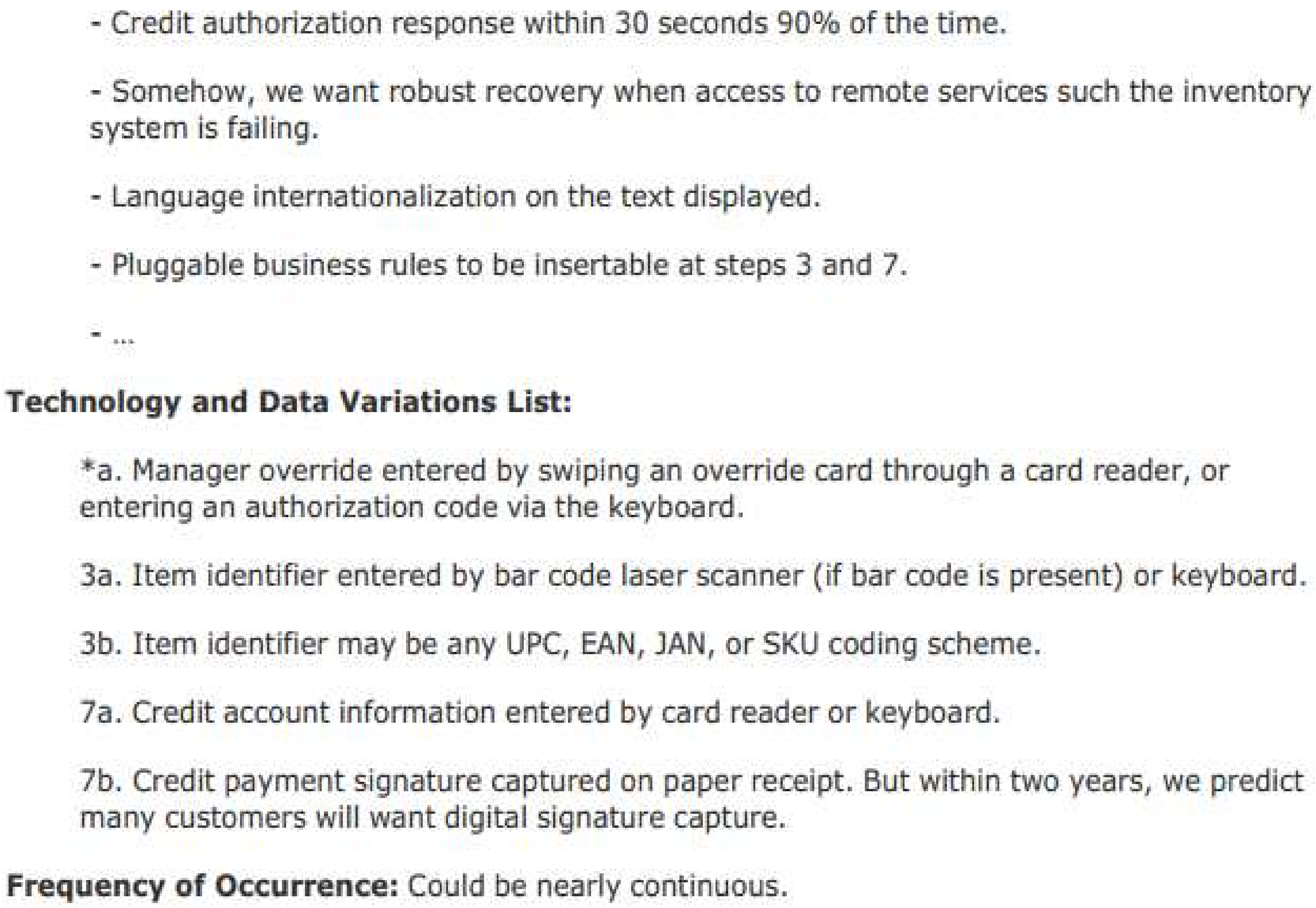
1. If System can detect the fault, will signal the problem.
2. Cashier replaces paper.
3. Cashier requests another receipt.

Special Requirements:

- Touch screen UI on a large flat panel monitor. Text must be visible from 1 meter.

Example: process sale, Fully dressed Style

Use case Example



- Credit authorization response within 30 seconds 90% of the time.

- Somehow, we want robust recovery when access to remote services such the inventory system is failing.

- Language internationalization on the text displayed.

- Pluggable business rules to be insertable at steps 3 and 7.

- ...

Technology and Data Variations List:

*a. Manager override entered by swiping an override card through a card reader, or entering an authorization code via the keyboard.

3a. Item identifier entered by bar code laser scanner (if bar code is present) or keyboard.

3b. Item identifier may be any UPC, EAN, JAN, or SKU coding scheme.

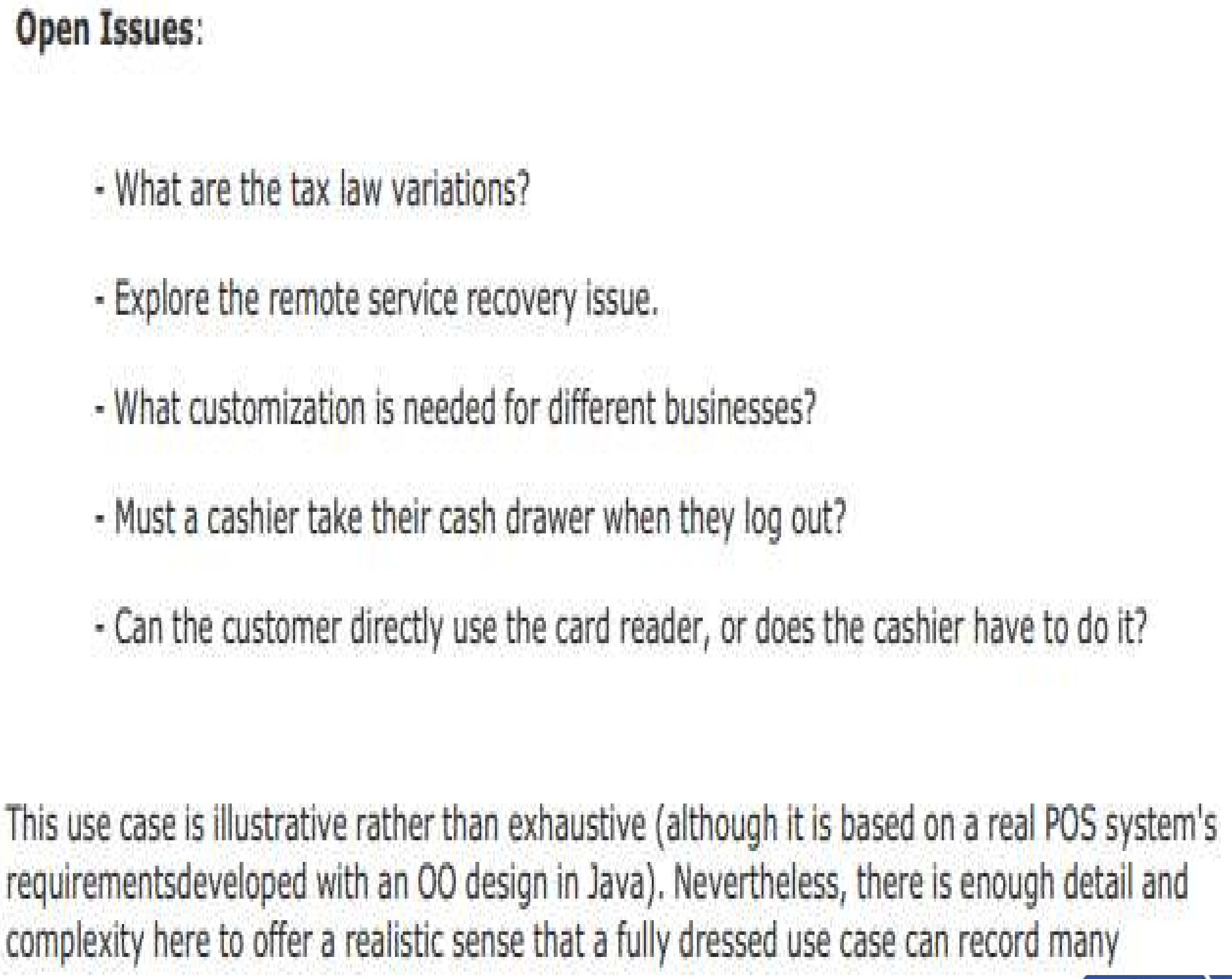
7a. Credit account information entered by card reader or keyboard.

7b. Credit payment signature captured on paper receipt. But within two years, we predict many customers will want digital signature capture.

Frequency of Occurrence: Could be nearly continuous.

Example: process sale, Fully dressed Style

Use case Example



Open Issues:

- What are the tax law variations?
- Explore the remote service recovery issue.
- What customization is needed for different businesses?
- Must a cashier take their cash drawer when they log out?
- Can the customer directly use the card reader, or does the cashier have to do it?

This use case is illustrative rather than exhaustive (although it is based on a real POS system's requirementsdeveloped with an OO design in Java). Nevertheless, there is enough detail and complexity here to offer a realistic sense that a fully dressed use case can record many