Chapter 4: Understanding Processes with Use Cases and Process Models

# Teaching Tips and Strategies *(from Alan Dennis)*

We have developed this book so that the focus is first on use cases, not DFDs, because in my consulting work, I've found it easier to start with the use cases, rather than jumping immediately into the DFDs. Use cases have fewer rules and are very close to how users think (and inexperienced analysts). There are no standards for the structure of a use case, so the form we developed is not "standard." Nonetheless, we have found it useful because it provides structure in helping the students to understand what questions to ask about processes and how to move from the big picture to the lower levels of detail. Following the form and the method described in the "building use cases" section also makes it very difficult for students to make common mistakes such as forgetting to having data go into data stores without a process moving them.

The process for building use cases (also not a standard) is a "breadth-first-search" technique; that is, the students gradually build each use case end-to-end, by repeatedly going through the use case gradually adding more and more detail to each element of it (rather than getting each part perfect before more on to the next part of the use case). I've found this to be the most effective approach for in my consulting practices when I've had users with little prior SA&D experience build use cases and I've simply pulled it over for use in the classroom. The key part in using this approach is to make students focus on each step as they do it, rather than jumping ahead. There is a temptation to try and do it all in one pass, but the problem is that everybody (myself included) makes mistakes in doing use cases, and the more times you iterate through the use case, the more likely you are to catch the mistakes and fix them before the project gets out of hand.

# Answer to Your Turn 4-1: Campus Housing

Student responses will vary. One example of a set of use cases:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Apt. Owner adds or deletes apartment | | ID: \_1\_ | Importance Level: High | |
| Primary Actor: Apt. Owner | | | | |
| Short Description: This use case describes how an apt. owner adds or deletes an apartment from the database of available apartments. | | | | |
| Trigger: Apt. owner has an apartment available or an apartment becomes unavailable  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Apartment owner name Apt. Owner  Apartment address Apt. Owner  Number of bedrooms Apt. Owner  Number of bathrooms Apt. Owner Monthly rent Apt. Owner  Restrictions Apt. Owner | Major Outputs:  Description Destination  Apartment owner name Apt.DataStore  Apartment address Apt.DataStore  Number of bedrooms Apt.DataStore  Number of bathrooms Apt.DataStore  Monthly rent Apt.DataStore Restrictions Apt.DataStore | | | |
| Major Steps Performed   1. Apartment owner contacts service    1. When apartment become available,       1. Apt. Owner fills out form with appropriate information.       2. Apartment Information is entered into database.    2. When apartment becomes unavailable:       1. Service deletes apartment | | | | Information for Steps  Apartment information  Completed Form  Apt info (datastore)  Request to delete apartment |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Student Database Search | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: Student | | | | |
| Short Description: This use case describes how a student searches the database for an apartment | | | | |
| Trigger: Student decides to search for an apartment  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Number of bedrooms Student  Number of bathrooms Student  Monthly rent Student  Restrictions Student | Major Outputs:  Description Destination Apartment Found Student  Apartment address Student  Number of bedrooms Student  Number of bathrooms Student  Monthly rent Student  Restrictions Student  Address Student  Owner Contact Info Student | | | |
| Major Steps Performed   1. Student browses to Apartment Search website. 2. Student enters search criteria. 3. Student contacts Apartment Owner(s) for selected apartments. | | | | Information for Steps  Website address  Student search criteria (bedrooms, bathrooms, rent, restrictions)  Search results  Search results Apartment Owner contact information |

# Answer to Your Turn 4-2: Functional Requirements for DrōnTeq Create a Parts Request

Student responses will vary. The functional requirement of 2.1, Create a Parts Request, in figure 3-3 of the DrōnTeq Customization Shop Management are recorded in the initial functional requirements. The student should be able to expand each of these requirements and provide an expanded revised list of functional requirements based on the UC-3 provided in Figure 4-10.

|  |
| --- |
| Initial Functional Requirement 2.1, Create a Parts Request, for DrōnTeq (from Figure 3-3) |
| * The system will send a Parts Request for needed drone components on an order to Drone Inventory department. (2.1). |
| Revised Functional Requirement 2.1 for DrōnTeq (based on UC-3, Figure 4-10) |
| * The shop manager initiates the creation of a Parts Request * For all needed components for the custom drone:   + The shop manager specifies the requirement component and quantity needed   + The system checks the quantity available for the part in the Inventory System and indicates that the needed component is in stock or out of stock with an expected availability date.   + The component is added to the Parts Request along with the availability info. * The shop manager specifies that all components have been entered into the Parts Request. * The system stores the new Parts Request * The system transmits the Parts Request to the Inventory Department |

# Answer to Your Turn 4-3: Functional Requirements for DrōnTeq Obtain Component Parts

Student responses will vary. The functional requirements of 2.3, Obtain Component Parts, in figure 3-3 of the DrōnTeq Customization Shop Management are recorded in the initial functional requirements. The student should be able to expand each of these requirements and provide an expanded revised list of functional requirements based on UC-4 and UC-5 provided in Figure 4-10.

|  |
| --- |
| Initial Functional Requirements for (from Figure 3-3) |
| * The system records the arrival of component parts as they arrive in the shop parts room. (2.3) |
| Revises Functional Requirements for DrōnTeq Obtain Component Part (based on UC-4, Figure 4-10) |
| * The Parts Room clerk retrieves the Parts Request using the Parts Request number. * The system requests that the salesperson confirms that the part is correct and is undamaged.   + If the part is not correct, the Part Request is closed; exit the use case   + If the part is damaged, the Part Request is closed; exit the use case   + If the part is correct and undamaged, the date/time of receipt is recorded * The Parts Room clerk enters the Parts Room location where the part is stored * The system stores the updated Parts Request datastore. * The system checks to see if all needed components on the Parts Request have been received from Inventory.   + If all parts have been received, the system begins the Finalize Parts Request process   + If not all parts have been received, the process ends. |

# Answer to Concepts in Action 4-A: Building a Bad System?

1. One reason the problems were missed is that customers were not considered as a source of inputs to the system. The agents were certainly documented, but no one realized that the customers might bypass the agents and directly use the system.
2. The analysts or agents might have researched similar sites to determine that customers could indeed use the system directly. The team might have conducted a walk-though with personnel other than agents and analysts, which may have brought the activity to light.
3. Instead of abandoning a system that worked fairly well, the company might have optimized it instead. Recognizing that customers use these types of sites, they might have made this so useful to the customer that the site, and the company name, became known as the best of place to go when searching for a home.

# Answer to Your Turn 4-4: Campus Housing

Student responses will vary depending on their answer to Your Turn 4-1.

# Solutions to End of Chapter Questions

1. *What is the purpose of developing use cases during systems analysis?*

A *use case* represents how a system interacts with its environment by illustrating the activities that are performed by the users of the system and the system’s responses, and is often thought of as an external or functional view of a business process. Use cases are developed during systems analysis activities to help the analysts better understand the situation and simplify later modeling steps in the analysis phase.

1. *How do use cases relate to the requirements stated in the requirements determination?*

A use case describes in more detail the key elements of the requirements definition. Use cases will provide more detail on the processes by which the system is to meet those requirements and the data the system needs to capture and store.

1. *Describe the elements of the use case’s basic information section.*

The basic information section includes use case name, number, a brief description, and the trigger for the use case.

1. *What is the purpose of the input and outputs section of the use case?*

The purpose of the inputs and outputs section is to list all inputs and associated sources, and outputs and associated destinations for the proposed system.

1. *What is the purpose of stating the primary actor for the use case?*

The purpose of stating the primary actor is to identify the external entity that initiates the event to which the system responds. The actor refers a person, another software system, or a hardware device that interacts with the system to achieve a useful goal.

1. *Why is it important to state the priority level for a use case?*

The priority level of a use case identifies the relative significance of the use case in the overall system. Use cases are classified as high, medium, or low. This classification allows for immediate identification of essential use cases.

1. *What is the distinction between an external trigger and a temporal trigger? Give two examples for each.*

A temporal trigger is based upon the passage of time. Examples are time to pay a bill; library book is due. External triggers are things outside the system to which the system must respond. Examples are a customer placing an order; customer payment is received.

1. *Why do we outline the major steps performed in the use case?*

Steps are outlined in order to determine the sequence of events that must occur to complete the use case. An outline of the sequence of events aids in ensuring that all steps are included.

1. *What is the purpose of an event-response list in the process of developing use cases?*

The purpose of an event-response list is to provide a step between the requirements definition and use cases and helps to clarify the number and scope of the use cases. Building an event-response list allows for a review of the requirements definition and associated actions.

1. *Should a use case be prepared for every item on the event-response list? Why or why not?*

Initially, every event on the event-list should be translated into a use case. The information in the functional requirements definition sometimes just flows into the use cases, but it usually requires some thought as to how to structure the use cases. Use case development can be viewed as an iterative process, and further analysis may allow the developer to discard or add use cases to the set.

1. *Describe two ways to handle a situation in which there are a large number of use cases.*

One possibility for the large number is that the use cases are not defined at the right level of detail. If the use cases are too ‘small’ we do not need to bother to create a use case, but simply use the information in the requirements definition itself to build the process and data models. If there really are more than eight or nine major use cases, the use cases are grouped together into packages of related use cases. These packages are then treated as the major processes for the top level of the process model with the use cases appearing on lower levels. The packages could also be treated as separate systems and modeled as separate systems.

1. *What role does iteration play in developing use cases?*

It is best to develop the major use cases and their basic information first so that no major events are forgotten. Then, the inputs and outputs are added and the detailed steps are outlined. As the inputs and outputs associated with each step are identified, more details may be identified. It is not practical to try and get every detail right the first time through the use case, so we cycle through the steps iteratively until the use case is complete.

1. *Describe the best way to validate the content of use cases.*

Role-playing can be useful in confirming the validity of the use case. If the users execute the steps of the use cases using the written steps of the use cases as a “script” for actions to take, they will be able to find errors or confirm the correctness of the use cases.

1. *What is a process model? What is a data flow diagram (DFD)? Are the two related? If so, how?*

Process models are used to document the information obtained during the analysis phase of a systems development project. DFD is a common technique used in processing modeling. They document the business processes and the data that passes between these business processes.

1. *Distinguish between logical process models and physical process models.*

Logical process models simply describe the process. They do not have any hint of how those processes are constructed. On the other hand, physical process models provide the information that the developer will need to build the system.

1. *Define what is meant by a process in a process model. How should a process be named? What information about a process should be stored in the CASE repository?*

A process is an activity or a function that is performed for some specific business reason. Every process should be named starting with a verb and ending with a noun. The information stored in the CASE repository should include the following:

* Number
* Name (verb phase)
* Description
* At least one output data flow
* At least one input data flow

1. *Define what is meant by a data flow in a process model. How should a data flow be named? What information about a data flow should be stored in the CASE repository?*

A data flow shows a single data or set of logically-related data items that move into or out of processes. It should be named using a now. The information stored in the CASE repository should include the following:

* Label (name)
* Type (Flow)
* Description
* Alias
* Composition (descriptions of the data elements)
* Any notes

1. *Define what is meant by a data store in a process model. How should a data store be named? What information about a data store should be stored in the CASE repository?*

A data store is a set of data that is stored all together. The data store is what actually contains or stores the data. As with a data flow, it should be named using a noun. The information regarding a data store that should be stored in the CASE repository includes the following:

* Label (name)
* Type (store)
* Description
* Alias
* Composition (a description of the data elements)
* Any notes

1. *Define what is meant by an external entity in a process model. How should an external entity be named? What information about an external entity should be stored in the CASE repository?*

An external entity is something outside the scope of the system but which interacts with the system. It could be a person, organization, or another system that supplies information to the system and/or receives information from the system. It should be named using a noun. The information regarding a data store that should be stored in the CASE repository includes the following:

* Label (name)
* Type (entity)
* Description
* Alias
* Any notes

1. *Why is a process model typically composed of a set of DFDs? What is meant by decomposition of a business process?*

Most business processes are simply too complex to be depicted using one single diagram. As a result, business processes are usually depicted with a set of DFFD. The first diagram is the Context Level and will show a summary of the system. Subsequent DFDs will show processes within that system. Decomposition means breaking down a business process into smaller, logical processes.

1. *Explain the relationship between a DFD context diagram and the DFD level 0 diagram.*

The first diagram is the Context Level and will show a summary of the system so it is a high-level overview of the entire system being represented by one process symbol. The only other things shown are data flowing into and out of the system along with the external entities that interact with the system.

A level 0 diagram is a high-level view of the internal process of the business system. Since they are at a high level, subsequent diagrams depict increasing levels of detail.

1. *Explain the relationship between a DFD level 0 diagram and DFD level 1 diagram(s).*

The DFD level 0 diagram only shows the major high-level processes of the system. Each process is then decomposed into a more detailed DFD level 1 diagram. Generally, there will be at least one DFD level 1 diagram for each process shown in the context level diagram.

1. *Discuss how the analyst knows how to stop decomposing the process model into more and more levels of detail.*

The complexity of the overall system being modeled drives this so it will be different for each system. Processes are usually decomposed when this additional decomposition helps to understand the process so this is driven by complexity. You should typically have at least three but not more than nine processes per DFD page.

1. *Suppose that a process on a DFD is numbered 4.3.2. What level diagram contains this process? What is this process’s parent process?*

Level 2 diagram contains the process numbered 4.3.2. Parent process is 4.3

1. *Explain the use of structured English in process descriptions.*

Structured English is one method of describing a process. Typically, short sentences are used to describe the elements or steps in the process.

1. *Why would one use a decision tree and/or decision table in a process description?*

Alternate methods of describing a process are a decision tree and a decision table. A decision tree is useful in that it aids in understanding decision logic pertaining to nodes (questions) and branches (answers). A decision table aids in understanding the actions (business policies) that based on a condition or a set of conditions.

1. *Explain the process of balancing a set of DFDs.*

Alternate methods of describing a process are a decision tree and a decision table. A decision tree is useful in that it aids in understanding decision logic pertaining to nodes (questions) and branches (answers). A decision table aids in understanding the actions (business policies) that based on a condition or a set of conditions.

Balancing means ensuring that all information presented in a DFD at one level is accurately represented in the next-level DFD. This does not mean that the information is identical, but that it is shown appropriately.

1. *How are mutually exclusive data flows (i.e., alternative paths through a process) depicted in DFDs?*

Mutually exclusive data flows (alternatives) are all shown on the data flow diagram. Nothing is shown on the diagram to indicate these are alternatives. The information in the written process description will document the fact that the data flows are mutually exclusive.

1. *Discuss several ways to verify the correctness of a process model.*

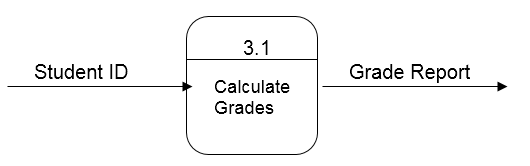
Model correctness can be verified by having the users validate it in a walk-through, or through role-playing. Analysts should also check that the decomposition of diagrams goes to the same level of detail (not necessarily the same number of levels). Analysts should also check for consistency in terminology throughout the model.

1. *Identify three typical syntax errors commonly found in DFDs.*

Students may list any of the syntax errors found in Figure 4-25.

1. *What is meant by a DFD semantic error? Provide an example.*

“Semantics” refers to the meaning of the DFDs and whether they accurately describe the business process being modeled. Semantics errors can be thought of as misunderstandings by the analyst in collecting, analyzing, and reporting information about the system. One example of a semantic error is one in which the data inputs to a process do not correspond to the type of information in the output data flow (see following DFD excerpt).



1. *Creating use cases when working with users is a recent development in systems analysis practice. Why is the trend today to employ use cases in user interviews or JAD sessions?*

Users may be confused or intimidated by the symbols used in drawing DFDs. Most users can verbalize the content of a use case more easily than they can grasp drawing DFDs. Use cases are easier to start with than DFDs for most users.

1. *How can you make a DFD easier to understand? (Think first about how to make one difficult to understand.)*

Try to ensure that the processes are clearly named. It is confusing if some processes on a diagram are significantly more complex than others on the same diagram. Try to structure the flow through the processes from top to bottom, left to right, since that is the natural way most Western cultures read diagrams.

*34. Suppose that your goal is to create a set of DFDs. How would you begin an interview with a knowledgeable user? How would you begin a JAD session?*

Although my goal is to create a set of DFDs, it is best to begin by creating a set of use cases. Therefore, I would begin my interviews or the JAD session by working with the users to identify the major events that are handled by the system and to complete the use cases for those events. After I had a complete and verified set of use cases, then I could work on structuring the information from them into a set of DFDs for the system.

# Solutions to End of Chapter Exercises

1. *Create a set of use cases for the process of buying glasses from the viewpoint of the patient.*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: See eye doctor | | ID: \_1\_ | Importance Level: High |
| Primary Actor: Patient | | | |
| Short Description: This describes how a patient obtains a prescription for new eyeglasses. | | | |
| Trigger: Patient requires new eyeglasses.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed | | | Information for Steps |

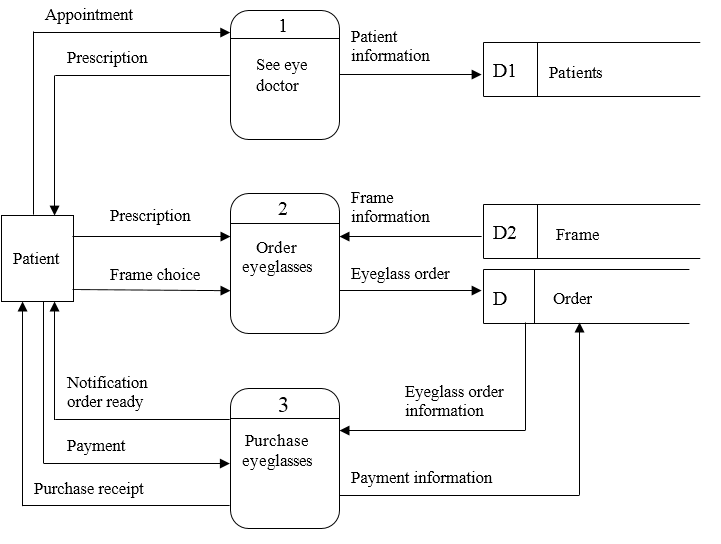
**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Select and order eyeglasses | | ID: \_2\_ | Importance Level: High |
| Primary Actor: Patient | | | |
| Short Description: This describes how a patient chooses and orders new eyeglasses. | | | |
| Trigger: Patient has new eyeglass prescription and visits eyeglass store.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed | | | Information for Steps |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Purchase glasses | | ID: \_3\_ | Importance Level: High | |
| Primary Actor: Patient | | | | |
| Short Description: This describes how a patient purchases new eyeglasses. | | | | |
| Trigger: New eyeglasses are ready and patient picks them up and pays for them.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed | | | | Information for Steps |

1. *Draw a level 0 data flow diagram (DFD) for the process of buying glasses in Exercise A.*



1. *Create a set of use cases for the following dentist office system…*

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Patient makes or changes an appointment | | ID: \_1\_ | Importance Level: High | |
| Primary Actor: Patient | | | | |
| Short Description: This describes how a new appointment is made or an existing appointment is changed. | | | | |
| Trigger: Patient calls to schedule new appointment or change an existing appointment.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed | | | | Information for Steps |

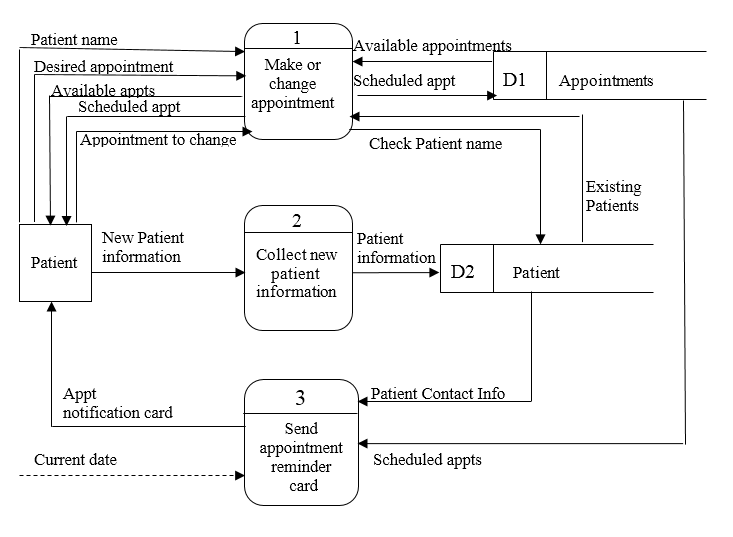
**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Collect new patient information | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: Patient | | | | |
| Short Description: This describes how new patient information is collected when a new patient arrives for appointment. | | | | |
| Trigger: New Patient arrives for appointment.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed | | | | Information for Steps |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Send appointment reminder card | | ID: \_3\_ | Importance Level: Medium | |
| Primary Actor: System | | | | |
| Short Description: This describes how reminder cards are sent to patients two weeks prior to appointment. | | | | |
| Trigger: Time to send reminders two weeks before scheduled appointments.  Type: External / **Temporal** | | | | |
| Major Inputs:  Description Source  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed | | | | Information for Steps |

1. *Draw a level 0 DFD for the dentist office system in Exercise C.*



1. *Complete the use cases for the dentist office system in exercise B by identifying the steps and the data flows within the use cases.*

**Use Case Description**

|  |  |  |
| --- | --- | --- |
| Use Case Name: Patient makes or changes an appointment | ID: \_1\_ | Importance Level: High |
| Primary Actor: Patient | | |
| Short Description: This describes how a new appointment is made or an existing appointment is changed. | | |
| Trigger: Patient calls to schedule new appointment or change an existing appointment.  Type: **External** / Temporal | | |
| Major Inputs:  Description Source  Patient name Patient  Desired appointment Patient  Appointment to change Patient  Available appointments Appointment file  Existing Patients\_\_\_\_ Patient file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Scheduled appointment Patient  Scheduled appt Appt file  Available appts Patient  Check Patient name Patient file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed   1. Get patient’s name. Check patient name with Patient file. 2. If new patient, get address and phone number and enter in new patient file entry. 3. If this is an appointment change, find current appointment in appointment file, and cancel it. 4. Get desired appointment time and match with available appointments in Appointment file. When a suitable time is found, enter scheduled appointment in Appointment file and confirm with patient. | Information for Steps  Patient name  Patient File  Address  Phone number  New Patient information  Appointment to change  Existing Appointments  Desired appointment  Available appointments  Scheduled appointments | |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Collect new patient information | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: New Patient | | | | |
| Short Description: This describes how new patient information is collected when a new patient arrives for appointment. | | | | |
| Trigger: New Patient arrives for appointment.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  New Patient information Patient  (Includes name, address,  Phone number and medical  History)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Patient Information Patient file    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. When the patient arrives, determine if he/she has been seen before in this office. If they have not, have them fill out a patient information form. 2. Use the information on the patient information form to update and complete the patient’s entry in the patient file. | | | | Information for Steps  Patient status  Patient name  Address  Phone number  Medical history |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Send appointment reminder card | | ID: \_3\_ | Importance Level: Medium | |
| Primary Actor: System | | | | |
| Short Description: This describes how reminder cards are sent to patients two weeks prior to appointment. | | | | |
| Trigger: Time to send reminders two weeks before scheduled appointments.  Type: External / **Temporal** | | | | |
| Major Inputs:  Description Source  Current date Calendar  Scheduled appts Appt file  Patient Contact Info\_\_\_ Patient file\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Appt notification card Patient  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. Each day, find the appointments in the Appointment file for the day two weeks ahead.   2. Prepare and mail appointment notification card to all patients having appointments. | | | | Information for Steps  Current date  Scheduled appointments  Name  Address  Appointment  Appt notification card |

1. *Create a set of use cases for an online university registration system…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain department course offerings | ID: \_1\_ | | Importance Level: High |
| Primary Actor: Department/Staff | | | |
| Short Description: This describes how department staff reviews course offerings, adds new courses, deletes existing courses or changes existing course information. | | | |
| Trigger: Departments must prepare upcoming course offerings.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Course offering changes Department Staff  Course offerings Course offering file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | | Major Outputs:  Description Destination  Updated course offerings Course  offering  file  Course offering list Dept staff  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed  1. Department staff requests course offering list for the department. List of course offerings is generated.   1. New course information is entered. 2. Courses to delete are entered. 3. Course modifications are entered. | | Information for Steps  Course offering list request  Department identifier  Course offering list  New course information  Course offering update  Course number to delete  Course offering update  Course number to modify  Course changes  Course offering update | |

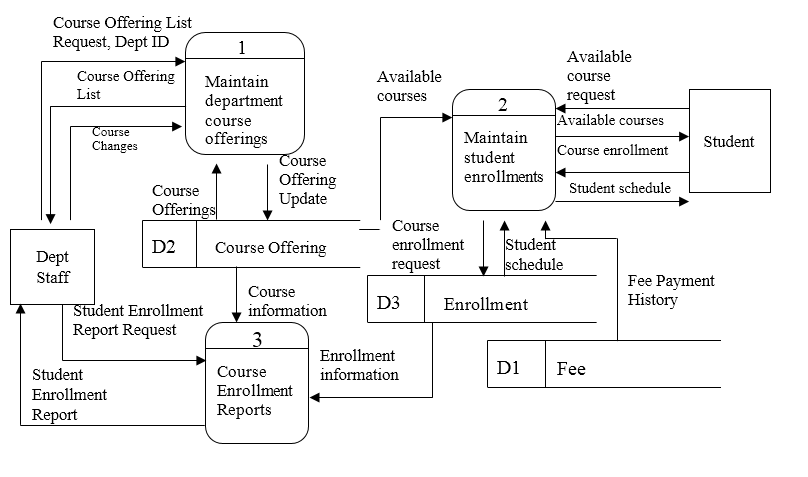
**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain student enrollments | ID: \_2\_ | | Importance Level: High |
| Primary Actor: Student | | | |
| Short Description: This describes how students review the listing of courses available for enrollment, add and remove courses from their schedules, and review their schedules. | | | |
| Trigger: Student needs to enroll in courses.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Available course request Student  Available courses Available course file  Course enrollment request Student  Fee payment status Fees file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | | Major Outputs:  Description Destination  Avail course list Student  Stdt enrlmt Enrlmt file  Student schedule Student  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed  1. Student requests list of available courses. List of available courses is generated.   1. Student adds course to current schedule. Fee payment status is checked and “total hours enrolled” is checked. If OK, course is added to student schedule. 2. Student removes course from schedule. 3. Student reviews current scheduled courses. | | Information for Steps  Available course request  Available course list  Course ID, Enrollment request  Stdt enrlmt  Fee payment history  Available courses Student Schedule  Course ID  Enrollment request  Student ID  Student schedule | |

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Course enrollment reports | ID: \_3\_ | | Importance Level: High |
| Primary Actor: Staff | | | |
| Short Description: This describes how department staff prints various reports on courses and enrollments. | | | |
| Trigger: Department staff needs information on courses and course enrollments.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Report request Staff  Course information Course offerings File  Enrollment information Enrollment File  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | | Major Outputs:  Description Destination  Report requested Staff  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed  1. Staff enters report request.   1. Requested report is generated. | | Information for Steps  Report request  Course offerings information  Enrollment information  Report requested | |

1. *Draw a level 0 DFD for the university system in Exercise F.*



1. *Create a set of use cases for A Real Estate, Inc….*

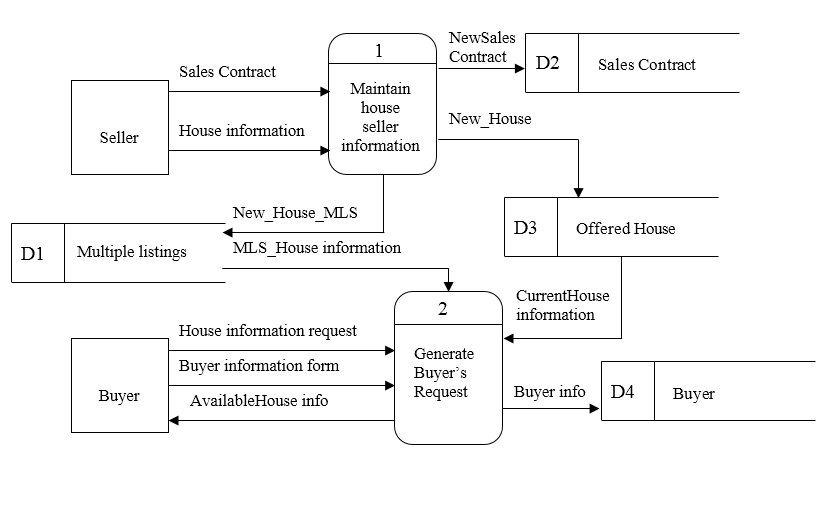
**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain house seller information | | ID: \_1\_ | Importance Level: High |
| Primary Actor: Seller | | | |
| Short Description: This describes how house sellers enter into a contract to sell and provide information on a house to sell. | | | |
| Trigger: House seller wishes to sell house.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Sales Contract Seller  House information Seller  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  NewSales Contract Contract file  New\_House Offered Houses file  New\_House\_MLS Multiple listings file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed  1. Seller signs contract and provides house information.   1. Offered House database is updated with new listing information. 2. Data on new listing is transmitted to Multiple Listings file. | Information for Steps  Sales contract  House information (seller)  House information (seller)  House information (new AREI listing)  House information (AREI Offered Houses)  House information (new MLS listing) | | |

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Generate Buyer’s Request | ID: \_2\_ | | Importance Level: High |
| Primary Actor: Buyer | | | |
| Short Description: This describes how buyers request and receive information on offered houses. | | | |
| Trigger: Buyers request information on available houses.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Buyer information form Buyer  House information request Buyer  CurrentHouse information Offered houses file  MLS\_House information Multiple listings file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | | Major Outputs:  Description Destination  AvailableHouse info Buyer  Buyer info Buyer file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed  1. If buyer requests information on a specific house, retrieve that information from the offered house file.   1. If buyer wants to search for several prospective houses, obtain a buyer information form from buyer. Search offered houses file and Multiple Listing file for houses matching buyers specifications and provide house information to buyer. | | Information for Steps  Buyer specific house search request  House request search results  Buyer information form  Buyer multiple house search request  House information search results | |

1. *Draw a level 0 DFD for the real estate system in Exercise H.*



1. *Create a set of use cases for a Video Store…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain Video inventory | ID: \_1\_ | | Importance Level: High |
| Primary Actor: Video Supplier | | | |
| Short Description: This describes how to add and remove videos from video database. | | | |
| Trigger: New video arrives or damaged videos are returned by customer.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Video descriptive information Video Supplier  Damaged video information Manager  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | | Major Outputs:  Description Destination  Video removed from Video file  stock  New video to rent Video file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | |
| Major Steps Performed  1. When new videos are received, enter video descriptive information into video file.   1. When damaged videos are returned, the video entry should be removed from the video file. | | Information for Steps  New video descriptive information  AVS video code & descriptive info  Damaged video information  AVS video code and delete request | |

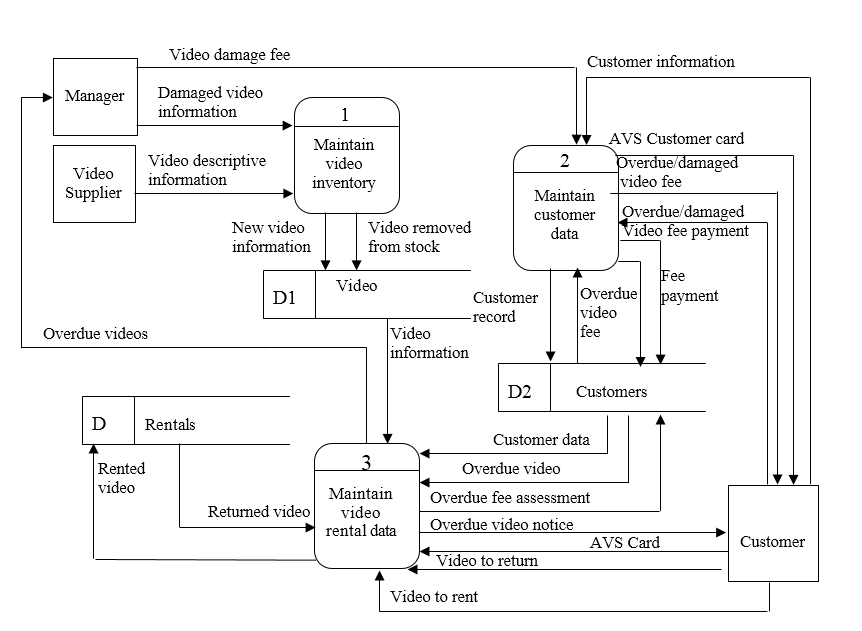
**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Maintain customer data | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: Customer | | | | |
| Short Description: This describes how customer information is established and overdue fine status is eliminated. | | | | |
| Trigger: Customer wants to rent a video.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Customer information Customer  Overdue/damaged video fee Customers file  Overdue/damaged fee payment Customer  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Customer record Customers file  AVS Customer card Customer  Fee payment Customers file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. When a new customer wants to rent a video, he/she provides customer information. A record is entered into the customer file and the customer is given an AVS customer card. 2. Customer removes overdue/damaged video fee by paying overdue fine. | | | | Information for Steps  Customer information  AVS Customer card  Customer information  Overdue video fee  Overdue fee payment |

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain video rental data | | ID: \_3\_ | Importance Level: High |
| Primary Actor: Customer | | | |
| Short Description: This describes how video rentals are entered and returns and overdue videos are recorded. | | | |
| Trigger: Videos are rented and returned and become overdue.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Video to rent Customer  AVS card Customer  Video to return Customer  Overdue video Customer file  Video information Video file  Customer data Customer file | Major Outputs:  Description Destination  Rented video Rentals file  Overdue fee assessment Customer file  Returned video Rentals file  Overdue video notice Customers  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed   1. Customer selects video to rent and provides AVS customer card. If there are no overdue videos and no unpaid overdue fees, the rental is entered in the rental file. 2. Customer returns rented video. 3. Video becomes overdue. An overdue fee is assessed and added to customer file. 4. Manager produces report and contacts customers with videos that are two or more days overdue. | | | Information for Steps  AVS Customer Card  Video to rent  Rented video  Overdue video fee  Returned video  Current Video rentals  Current date  Video due date  Overdue fee assessment  Overdue video notice  Current Date  Rented and overdue videos  Customer information for overdue rentals |

1. *Draw a level 0 DFD for the video store system in Exercise J.*



1. *Create a set of use cases for the following health club membership system…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain members | | ID: \_1\_ | Importance Level: High |
| Primary Actor: Member | | | |
| Short Description: This describes how club records new member information, makes member changes to member information, and removes members. | | | |
| Trigger: Need to add, modify, or delete members.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Member information Member  Member information updates Member  Member to delete Management  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  New member record Members file  Member updates Members file  Member to delete Members file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed  1. New member joins club and provides descriptive member information.   1. Existing member provides updates to member information. 2. Members whose memberships have expired and who will not renew plus members who have died or moved away are removed from member file. | Information for Steps  New member information  Current membership information  Member information changes  Member information updates  Current membership information  Member to delete  Current membership information | | |

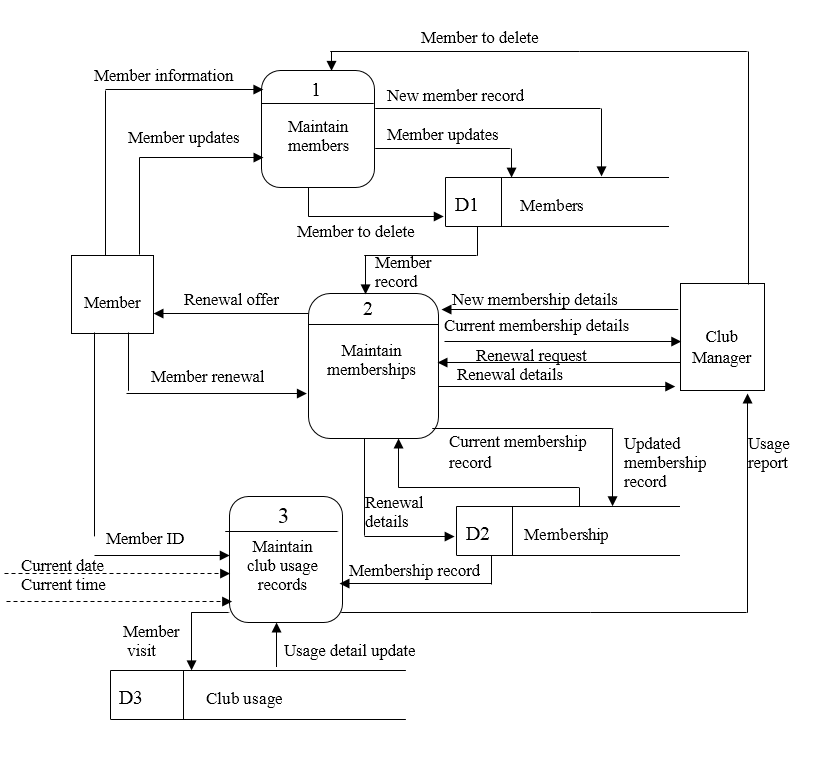
**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Maintain memberships | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: Manager | | | | |
| Short Description: This describes how to record new memberships and process membership renewals. | | | | |
| Trigger: Need to record memberships and renewals.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  New membership details Manager  Renewal request Manager  Member record Members file  Current Membership record Membership file  Member renewal Member | Major Outputs:  Description Destination  Updated Membership record Membership file  Renewal offer Member  Renewal details Manager  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. When new membership is established, manager provides details of agreement (length and fee). 2. Contact members one month prior to membership expiration and offer renewal terms. 3. Current member renews membership. | | | | Information for Steps  Membership details  Membership record  Current Date  Current membership details  Renewal details  Renewal offer  Member information  Renewal information  Membership details |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Maintain club usage records | | ID: \_3\_ | Importance Level: High | |
| Primary Actor: Member | | | | |
| Short Description: This describes how to record and report on member usage of the club. | | | | |
| Trigger: Need to record member visit to club and produce manager's reports.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Member ID Member  Current date Calendar  Current timeCalendar  Membership record Membership file  Usage detail update Club Usage File | Major Outputs:  Description Destination  Member visit Club usage file  Usage report Manager  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. Member checks in when using club. Member ID, time, and date of visit are recorded. 2. Manager wants report of heavy users and report of inactive members. | | | | Information for Steps  Current Date  Current Time  Member ID  Visit date  Visit time  Report request  Member visit records  Member information |

1. *Draw a level 0 DFD for the health club system in Exercise L.*



1. *Create a set of use cases for the Picnics-R-Us catering system…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Establish picnic contracts | | ID: \_1\_ | Importance Level: High |
| Primary Actor: Customer | | | |
| Short Description: This describes how customers receive information and book picnics. | | | |
| Trigger: Customer needs to establish picnic contract.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Picnic request Customer  Standard Menu Menu file  Special Requests Customer  Special pricing details Owner  Customer Details Customer | Major Outputs:  Description Destination  Standard menus Customer  Special requests Owner  Special pricing info Customer  Initial picnic Contract file  contract  Customer information Customer file | | |
| Major Steps Performed  1. Customer calls with picnic request and receives standard menu.   1. Customer has special request and received owner's cost estimate. 2. Customer decides to book picnic and an unconfirmed picnic contract is recorded in contract file. | Information for Steps  Picnic request  Standard menus  Special request  Special pricing  Customer details  Unconfirmed picnic contract | | |

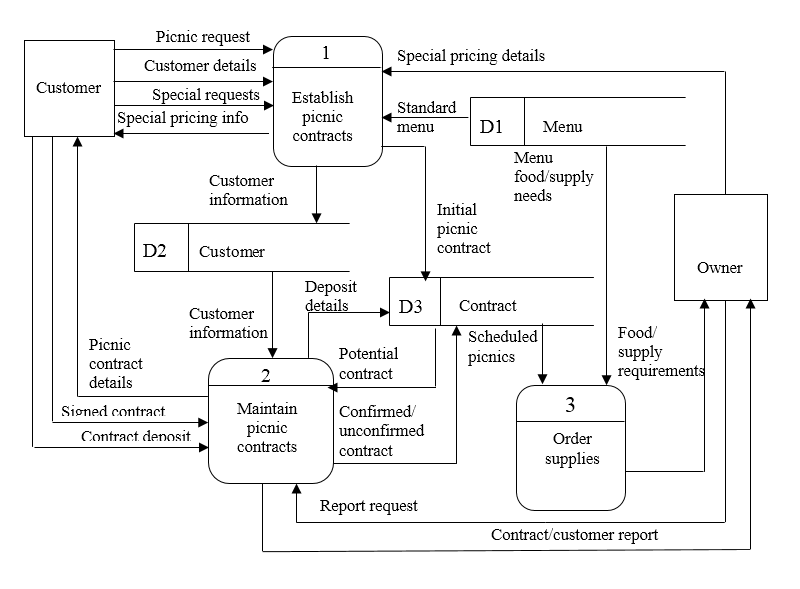
**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Maintain picnic contracts | | ID: \_2\_ | Importance Level: High | |
| Primary Actor: Contracts | | | | |
| Short Description: This describes how customer receives contract information and may confirm contract. | | | | |
| Trigger: Picnic contracts must be communicated to and confirmed by customers.  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Potential contract Contract file  Signed contract Customer  Contract deposit Customer  Report request Owner  Customer information Customer file | Major Outputs:  Description Destination  Confirmed/Unconfirmed  contracts Contract file  Deposit details Contract file  Contracts/customer report Owner  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. Customer is sent contract details. 2. Customer returns signed contract with deposit information. 3. Owner requests reports for marketing purposes | | | | Information for Steps  Unconfirmed picnic contract  Signed contract  Contract deposit information  Confirmed picnic contract  Report request  Picnic Contracts information  Customer information  Requested reports |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Order supplies | | ID: \_3\_ | Importance Level: Medium | |
| Primary Actor: System | | | | |
| Short Description: This describes how owner reviews upcoming picnics and orders required supplies. | | | | |
| Trigger: Time to prepare for next weekend's picnics (weekly).  Type: External / **Temporal** | | | | |
| Major Inputs:  Description Source  Scheduled picnics Contracts file  Menu food/supply needs Standard menu file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Food/supply requirements Owner  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. Identify picnics for upcoming weekend. 2. Determine food and supply requirements by reviewing upcoming weekend’s picnics, picnic menus, and food/supply inventory. | | | | Information for Steps  Current Date  Scheduled Picnics  Weekend Picnics  Weekend Picnics  Standard menus  Standard menu food and supply needs  Food qty-on-hand  Supply qty-on-hand |

1. *Draw a level 0 DFD for the Picnics R Us system in Exercise N.*



1. *Create a set of use cases for the Of-the-Month-Club…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain club memberships | | ID: \_1\_ | Importance Level: High |
| Primary Actor: Customer | | | |
| Short Description: This describes how to create, change, and delete members. | | | |
| Trigger: Customer wants to enroll in one or more club memberships.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Member information Member  Member updates Member  Member to delete Management | Major Outputs:  Description Destination  Member details Members file  Updated member Members file  Deleted member Members file | | |
| Major Steps Performed  1. Customer calls and wishes to establish club membership. Member details are recorded.   1. Customer needs to change member details (such as address). 2. Remove expired member from the club records. | Information for Steps  New member request  Member details  Member information changes  Member updates  Member to delete | | |

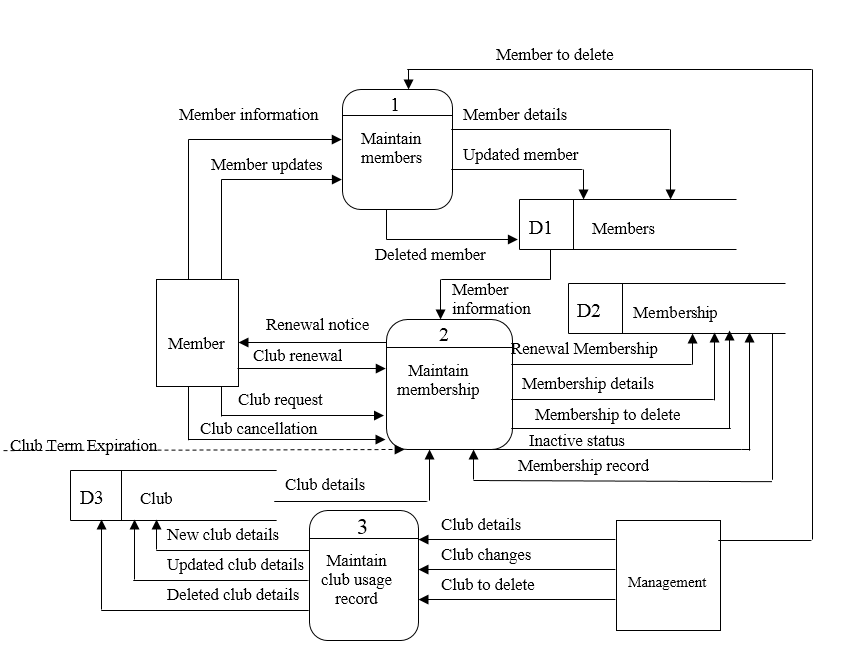
**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain memberships | | ID: \_2\_ | Importance Level: High |
| Primary Actor: Member | | | |
| Short Description: This describes how to add, renew, or delete memberships. | | | |
| Trigger: Customer enrolls in one or more clubs.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Club request Member  Club details Club file  Club renewal Member  Club term expiration Calendar  Club cancellation Member  Membership details Membership file | Major Outputs:  Description Destination  Membership record Membership file  Renewal notice Member  Renewal membership Membership file  Membership to delete Membership file  Inactive status\_\_\_ Membership file\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed   1. Member selects one or more clubs to join and membership in the club(s) is established. 2. Member is sent renewal notice when end of club term nears. 3. Member renews or cancels membership. 4. Membership expires without being renewed. | | | Information for Steps  Club request  Club details  Membership record  Current Date  Membership details  Club term expiration  Renewal notice  Club renewal or cancellation  Club details  Membership details  Renewal membership  Membership to delete  Membership details  Current Date  Inactive membership status |

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain club | | ID: \_3\_ | Importance Level: Medium |
| Primary Actor: Management | | | |
| Short Description: This describes how clubs are added, changed. or deleted. | | | |
| Trigger: Management wants to establish, change, or eliminate clubs.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Club details Management  Club changes Management  Club to delete Management  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  New club details Clubs file  Updated club details Clubs file  Deleted club details Clubs file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed   1. Management establishes new club. 2. Management wishes to change features of a club. 3. Management wishes to eliminate a club. | | | Information for Steps  Club details  Club updates  Club to delete |

1. *Draw a level 0 DFD for the Of-the-Month Club system in Exercise P.*



1. *Create a set of use cases for a university library borrowing system…*

**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Maintain books | | ID: \_1\_ | Importance Level: Medium |
| Primary Actor: Management | | | |
| Short Description: This describes how to record owned books and books that are lost or removed due to damage. | | | |
| Trigger: Books are to be added or removed from holdings.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Book details Manager  Books to remove Manager  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  New book entry Book holdings  Book to delete Book holdings  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed   1. New books are added to book holdings. 2. Lost or damaged books are removed from book holdings. | Information for Steps  Book details  New book entry  Book(s) to remove  Book to delete | | |

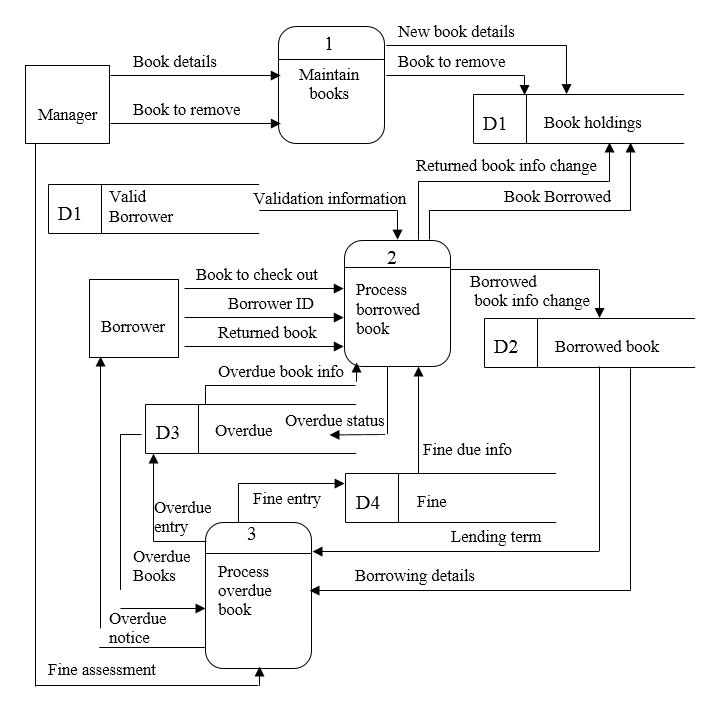
**Use Case Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name: Process borrowed book | | ID: \_2\_ | Importance Level: High |
| Primary Actor: Borrower | | | |
| Short Description: This describes how to check out and return books. | | | |
| Trigger: Borrower wishes to check out or return book.  Type: **External** / Temporal | | | |
| Major Inputs:  Description Source  Book to check out Borrower  Borrower ID Borrower  Validation information Valid Borrower file  Overdue book info Overdue file  Fine due info Fines file  Returned book Borrower  Borrowed book info Borrowed book file | Major Outputs:  Description Destination  Borrowed book info change Borrowed book file  Returned book info change Book holdings  Overdue status Overdue file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Major Steps Performed   1. Borrower presents book(s) to check out. ID is checked for validity against valid borrower file. Overdue books and fines are checked. If OK book is lent to borrower. 2. Borrower returns book. Book is removed from borrowed file and returned to holdings. If overdue, book is removed from overdue file. | | | Information for Steps  Book to check out  Borrower ID  Validation information  Overdue books info  Fines due info  Borrowed book info  Returned book  Borrowed book info  Borrowed book info change  Returned book info change  Overdue file status |

**Use Case Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Process overdue books. | | ID: \_3\_ | Importance Level: High | |
| Primary Actor: System | | | | |
| Short Description: This describes how to identify overdue books, notify borrower, assess fines. | | | | |
| Trigger: Book lending term expires.  Type: External / **Temporal** | | | | |
| Major Inputs:  Description Source  Lending term Borrowed book file  Borrowing details Borrowed book file  Fine assessment Manager  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | Major Outputs:  Description Destination  Overdue entry Overdue file  Overdue notice Borrower  Fine entry Fines file  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| Major Steps Performed   1. Borrowed book lending term expires. Entry is made in overdue file. 2. Borrowed book is more than two weeks overdue. Fine is assessed. 3. Every week, overdue book notices are sent to borrowers. 4. Manager specifies fine for lost or damaged book. | | | | Information for Steps  Current date  Book lending term  Overdue status change  Borrowing details  Fine entry  Borrowing details  Overdue book notice  Book damage entry  Fine entry |

1. *Draw a level 0 DFD for the university library system in Exercise R.*



# Answers to Textbook Minicases

*1. a. The Analysis phase of systems development has three primary goals: (1) to understand the current system (called the As-Is system); (2) to identify ways to improve the situation; and (3) to develop a conceptual design for the new system (called the To-Be system). To achieve these goals, we will be spending time talking to each of you to learn about how the current system works, the problems you experience, the improvements you’d like to see in the new system, and your ideas for what the new system should be like.*

*b. To help us understand your business processes, we utilize a tool called a use case. The use case represents the activities that are performed by users of your system and the information needed to accomplish these activities. The use case is prepared from the perspective of the client, and we will be working closely with you and your staff to understand the essential activities of your system. We will meet with you and your staff several times to refine our use cases and to add appropriate details as our understanding of necessary activities and information grows richer. Our use cases will form the foundation of our next steps in analyzing your current system and designing the new system.*

*Use cases are used to assist the users to represent the business processes in an informal, verbal way rather than with the formality of data flow diagrams. This technique can be used to help develop process models of both the As-Is and the To-Be systems. Use cases use a three-stage process that develops the information needed to create data flow diagrams. First, the users identify the major activities that the users perform (referred to as use cases). Then the team works through several steps to develop a more detailed outline of the activities performed within each use case and the inputs and outputs needed to perform each step.*

2. There are multiple semantic and syntactic errors in the level 1 DFD for the Hatcher Company inventory. Errors include:

1. Processes with input, but no output = 5.3 Reconcile Shipment
2. Processes with output, but no input = 5.4 Notify Accounts Payable
3. Entity to entity dataflow = Warehouse Manager to Supplier
4. Same data flow name on two separate dataflows = New inventory
5. Data flow sent directly to external entity from data store = Inventory received from D2 Inventory on Hand to Warehouse manager
6. Physical tasks were depicted = unpacking inventory and stocking shelves.

One possible solution is to remove the Warehouse entity, remove processes 5.1 and 5.2 and revamp the processes.



3. A potential level 1 diagram for this problem is found here:

A close up of a map

Description automatically generated

4.Use cases:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Validate Re quest for Staffing | | ID: \_1\_ | Importance Level: High | |
| Primary Actor: Client | | | | |
| Short Description: This describes how a client staffing request is validated | | | | |
| Trigger: Request for staff arrives from client  Type: **External** / Temporal | | | | |
| Major Inputs:  Description Source  Staffing request Client  Contract terms Client contracts  file | Major Outputs:  Description Destination    Contract No. Client contracts file  Denial letter Client  Denial letter copy Denial Letter File  Valid staffing request Find Staff  Open staffing request Staffing requests file | | | |
| Major Steps Performed   1. Obtain the contract number from the staffing request. 2. Use the contract number to find the client contract in contract file. 3. Determine if contract is still valid and covers the type of staff requested. 4. If Request is valid, place request info in open staffing requests and route staffing request to Find Staff process. 5. Otherwise, prepare denial letter to send to client along with original staffing request. File denial letter. | | | | Information for Steps  Staffing request  Contract number  Contract terms  Staffing request  Staffing request  Denial letter. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case Name: Find Staff | | ID: \_2\_ | Importance Level: Medium | |
| Primary Actor: Staffing Request Approval | | | | |
| Short Description: Qualified staff are identified and reserved for a staffing request | | | | |
| Trigger: Valid Staffing request is received.  Type: **External** / **Temporal** | | | | |
| Major Inputs:  Description Source  Valid Staffing Request Validate Staffing  Qualified Staff Staff File | Major Outputs:  Description Destination  Reserved Staff Staff File  Unable to fill memo Complete Placement  Valid Request Complete Placement | | | |
| Major Steps Performed   1. Use staffing request details to identify potential qualified staff. 2. If one or more qualified staff are identified, mark them reserved in the Staff file. 3. If no qualified staff are found, prepare an Unable to fill memo. 4. Send staffing request (and unable to fill memo, if any) to Complete Placement. | | | | Information for Steps  Staff file details  Staff file  Staffing request details  Staff request, unable to fill memo |

# Supplemental Minicases

1. Refer to Minicase #1 from Chapter 4 of the textbook. Following your introductory comments and your attempts to explain the work that will be performed during the analysis phase of this project, you can feel a sense of uneasiness beginning to creep into the staff’s demeanor. One staff member was heard to comment to another during the break, “I came in to this meeting this morning all excited about this project and the new system we’ll be getting. Now I feel scared about it. I don’t know how I can help develop all those complicated models and diagrams. Sure I know how things work around here, but trying to draw pictures of our processes just confuses me.”

After overhearing this comment, you quickly decide that you will want to employ use cases in this situation to assist the staff in developing the system’s process models. In clear, non-technical terms, explain the process of developing use cases and how they will be used by the team in the development of logical process models for this system.

*Answer: Use cases are used to assist the users to represent the business processes in an informal, verbal way rather than with the formality of data flow diagrams. This technique can be used to help develop process models of both the As-Is and the To-Be systems. Use cases use a three-stage process that develops the information needed to create data flow diagrams. First, the users identify the major activities that the users perform (referred to as use cases). Then the team works through several steps to develop a more detailed outline of the activities performed within each use case and the inputs and outputs needed to perform each step.*

# Experiential Exercises

1. Purpose: To experience the benefits of validating use cases.

Divide the class into groups of four students. Within each group, have the students work in pairs. Have each pair of students work on developing the solution to one of the end-of-chapter exercises in the textbook. When they have completed these use cases, have the pairs within each group exchange use cases. Have each pair try to walk through the description developed by the other pair. Anything that is unclear or that was omitted should be discussed among the group to identify problems and weaknesses in their use cases.