Problem 13.22

(Term Project) Perform the analysis workflow of the Chocoholics Anonymous product described in Appendix A.

Step-by-step solution

1. **Step 1** of 1

If a provider resigns, the information about him cannot be deleted from the system before all his claims have been processed.  Similarly, the information about a member or a service cannot be deleted until all claims concerning the member or service have been processed.  Thus such deletions should be delayed until the end of the week.  This requirement has been ignored in the analysis workflow since the timing of actions is not a concern of analysis.  A “nice to have” requirement is that members should be informed of new providers, resignation of providers and changed provider details.  Similarly, providers should be informed of new services, discontinuation of services and changed service details.  The requirements workflow includes sending such information by email to the providers and members.  This has been left out of the analysis workflow since it is not essential to the product.

Functional Modeling

 Scenarios for the use cases of Chocoholics Anonymous depicted in Figure 11.17 (b) to (d) appear in Figures 13.88 to 13.96.

|  |
| --- |
| A provider wishes to initiate a session to use the ChocAn system to verify members, submit claims and receive a provider directory.  The provider switches on his or her terminal.    1. The system prompts the provider for his or her provider number.    2.  The provider supplies his or her provider number.    3. The system searches for a provider with this number.    4. The system displays a message indicating the options available to the provider.    The provider can now use the system.    5. When the provider has finished using the system, he or she chooses the Quit option.    6. The system ends the session. |
| **Possible Alternatives**    A.           In step 3, the system cannot find the provider number.  The system displays the message Invalid number, and step 1 is repeated. |

Figure 13.88.  An extended scenario for the Manage Session use case.

|  |
| --- |
| A provider wishes to verify that the number on a member’s card is a valid ChocAn membership number of an active member.    The provider has switched on his or her terminal and entered his or her provider number correctly.    1.  The provider swipes the member’s card through the card reader of the provider terminal.    2. The system searches for a member with this number.    3 The system displays a message indicating the status of the member on the provider terminal’s display, namely, Validated. |
| **Possible Alternatives**    A.           In step 2, the system cannot find the member number.  The system displays the message Invalid number, and the use case terminates.    B.           In step 3, the member’s status is “suspended.”  The system displays the message Suspended, and the use case terminates. |

Figure 13.89.  An extended scenario for the Verify Member use case.

|  |
| --- |
| A provider wishes to submit a claim for a service provided to a ChocAn member.  He or she has already verified the member as in the normal scenario of the Verify Member use case (Figure 13.88)    1. The provider enters the date the service was provided in the format MM-DD-YYYY, the service code and optionally comments about the service rendered.    2. The system verifies that the service code exists.  The system displays the corresponding service name.    3. The provider confirms that this is the correct service rendered.    4. The system stores the following information about the claim submitted:    Current date and time (MM-DD-YYYY HH:MM:SS)    Service date (MM-DD-YYYY)    Provider number (max 9 digits)    Member number (max 9 digits)    Service code (max 6 digits)    5. The system displays the fee to be paid to the provider for the service rendered.    The provider has a form on which he or she can enter the same information as that stored by the system, as well as the service fee. |
| **Possible Alternatives**    A.               In step 2, the service code does not exist.  The system displays the message Invalid service code.  The provider re-enters the service code and step 2 is repeated.    B.               In step 3, the service name is not the name of the service for which the provider wishes to submit a claim.  The provider re-enters the service code and step 2 is repeated. |

Figure 13.90.  An extended scenario for the Submit Claim use case.

|  |
| --- |
| The provider has already switched on his or her terminal and entered his or her provider number correctly.    1. The provider submits a request for a Provider Directory through his or her terminal.    2. The system generates a list (report) of all services, ordered alphabetically according to the service name, including, for each service:    Service name (max 20 characters)    Service code (max 6 digits)    Service fee (max $999.99)    3. The system sends the list as an email attachment to the provider, and displays a message to this effect on the provider’s terminal. |

Figure 13.91.  Scenario for the Receive Provider Directory use case.

|  |
| --- |
| A ChocAn operator wishes to add a new member.    1. The ChocAn operator selects the option to add a new member, and enters the new member’s details:    Member name (max 25 characters)    Member street address (max 25 characters)    Member city (max 14 characters)    Member state (2 letters)    Member zip code (5 digits)    Member email address (max 50 characters)    2. The system allocates a number to the new member, records the new member’s details, and sets the new member’s status to Active.    3. The system displays the new member’s member number and other details. |

Figure 13.92(a).  First scenario for the Maintain Member use case (add a new member).

|  |
| --- |
| A ChocAn operator wishes to update an existing member’s details.    1. The ChocAn operator enters the existing member’s member number.    2. The system searches for the details for the member and displays the details.    3. The ChocAn operator edits the details that must be changed.  (The member number can never be changed.)    4. The system updates the member’s details. |
| **Possible Alternative**    A. In step 2, the system cannot find the provider number.  It displays an error message, and the use case is terminated. |

**Figure 13.92(b).  Second extended scenario for the Maintain Member use case (update an existing member).**

|  |
| --- |
| A ChocAn operator wishes to delete an existing member.    1. The ChocAn operator enters the existing member’s member number.    2. The system searches for the details for the member and displays the details.    3. The ChocAn operator selects the option to delete the member.    4. The system deletes the member’s details. |
| **Possible Alternative**    A. In step 2, the system cannot find the provider number.  It displays an error message, and the use case is terminated. |

Figure 13.92(c).  Third extended scenario for the Maintain Member use case (delete an existing member).

|  |
| --- |
| A ChocAn operator wishes to add a new provider.    1. The ChocAn operator selects the option to add a new provider, and enters the new provider’s details:    Provider name (max 25 characters)    Provider street address (max 25 characters)    Provider city (max 14 characters)    Provider state (2 letters)    Provider zip code (5 digits)    Provider email address (max 50 characters)    Provider type (Dietitian, Internist or Exercise Specialist)    2. The system allocates a number to the new provider and records the new provider’s details.    3. The system displays the new provider’s provider number and other details. |

Figure 13.93(a).  First scenario for the Maintain Provider use case (add a new provider).

|  |
| --- |
| A ChocAn operator wishes to update an existing provider’s details.    1. The ChocAn operator enters the existing provider’s provider number.    2. The system searches for the details for the provider and displays the details.    3. The ChocAn operator edits the details that must be changed.  (The provider number can never be changed.)    4. The system updates the provider’s details. |
| **Possible Alternative**    A. In step 2, the system cannot find the provider number.  It displays an error message, and the use case is terminated. |

Figure 13.93(b).  Second extended scenario for the Maintain Provider use case (update an existing provider).

|  |
| --- |
| A ChocAn operator wishes to delete an existing provider.    1. The ChocAn operator enters the existing provider’s provider number.    2. The system searches for the details for the provider and displays the details.    3. The ChocAn operator selects the option to delete the provider.    4. The system deletes the provider’s details. |
| **Possible Alternative**    A. In step 2, the system cannot find the provider number.  It displays an error message, and the use case is terminated**.** |

Figure 13.93(c).  Third extended scenario for the Maintain Provider use case (delete an existing provider).

|  |
| --- |
| A ChocAn operator wishes to add a new service.    1. The ChocAn operator selects the option to add a new service, and enters the new service’s details:    Service code (max 6 digits)    Service name (max 20 characters)    Service fee (max $999.99)    2. The system records the new service details.    3. The system displays the new service details. |

Figure 13.94(a).  First scenario for the Maintain Service use case (add a service)

|  |
| --- |
| A ChocAn operator wishes toupdate an existing service.    1. The ChocAn operator enters the existing service code.    2. The system searches for the details for the service and displays the details.    3. The ChocAn operator edits the details that must be changed.    4. The system updates the service details. |
| **Possible Alternative**    A. In step 2, the system cannot find the service code.  It displays an error message, and the  use case is terminated. |

Figure 13.94(b).  Second extended scenario for the Maintain Service use case (update a service)

|  |
| --- |
| A ChocAn operator wishes todelete an existing service.    1. The ChocAn operator enters the existing service code.    2. The system searches for the details for the service and displays the details.    3. The ChocAn operator selects the option to delete the service.    4. The system deletes the service details. |
| **Possible Alternative**    A. In step 2, the system cannot find the service code.  It displays an error message, and the  use case is terminated. |

Figure 13.94(c).  Third extended scenario for the Maintain Service use case (delete a service).

|  |
| --- |
| The ChocAn manager wishes to print a provider report.    1. The ChocAn manager selects the provider report, and enters the end date of the week he or she requires and the provider number.    2. The system generates a provider report including the following information:    Provider name (max 25 characters)    Provider number (max 9 digits)    Provider street address (max 25 characters)    Provider city (max 14 characters)    Provider state (2 letters)    Provider zip code (5 digits)    For each service provided, the following information, sorted according to claim submission  date and time, is included:     Service date (MM-DD-YYYY)     Claim submission date and time (MM-DD-YYYY HH:MM:SS)     Member name (max 25 characters)     Member number (max 9 digits)     Service code (max 6 digits)     Service fee (max $999.99)    Total number of consultations with members (3 digits)    Total fee for week (max $99,999.99)    3. The manager chooses to print the report.    4. The system prints the report. |
| **Possible Alternatives**    A.               In step 1, the ChocAn manager enters an invalid provider number.  The system displays an error message and the use case terminates.    B.               In step 3, the ChocAn manager chooses not to print the report.  The use case terminates. |

Figure 13.95(a).  First extended scenario for the Request Report use case (provider report).

|  |
| --- |
| The ChocAn manager wishes to print a member report.    1. The ChocAn manager selects the member report, and enters the end date of the week he or she requires and the member number.    2. The system generates a member report including the following information.    Member name (max 25 characters)    Member number (max 9 digits)    Member street address (max 25 characters)    Member city (max 14 characters)    Member state (2 letters)    Member zip code (5 digits)    For each service provided, the following information, sorted according to service date, is  included:     Service date (MM-DD-YYYY)     Provider name (max 25 characters)     Service name (max 20 characters)    3. The manager chooses to print the report.    4. The system prints the report. |
| **Possible Alternatives**    A.               In step 1, the ChocAn manager enters an invalid member number.  The system displays an error message and the use case terminates.    B.               In step 3, the ChocAn manager chooses not to print the report.  The use case terminates. |

Figure 13.95(b).  Second extended scenario for the Request Report use case (member report).

|  |
| --- |
| The ChocAn manager wishes to print an accounts payable report.    1. The ChocAn manager selects the accounts payable report, and enters the end date of the week he or she requires.    2. The system generates an accounts payable report including the following information:    For each provider to be paid that week:     Provider name (25 characters)     Number of consultations (max 6 digits)     Total fee (max $99,999.99)    Total number of providers who provided services (max 6 digits)    Total number of consultations (max 9 digits)    Overall total fee (max $999, 999.99)    3. The manager chooses to print the report. |
| **Possible Alternatives**    A. In step 3, the ChocAn manager chooses not to print the report.  The use case terminates. |

Figure 13.95(c).  Third extended scenario for the Request Report use case (accounts payable report).

|  |
| --- |
| The accounting procedure is run every week at midnight on Friday.    1. For each provider, the system generates a provider report as in Figure 13.95(a) (first extended  scenario for the Request a Report use case), and sends the report as an email  attachment to the provider.    2. For each member, the system generates a member report as in Figure 13.95(b) (second  extended scenario for the Request a Report use case), and sends the report as an email  attachment to the member.    3. The system generates an accounts payable report for the ChocAn manager as in Figure 13.95(c) (third extended scenario for the Request a Report use case), and sends the report to the manager as an email attachment.    4. The system generates the following EFT data for each provider who must be paid, for the week:    Provider name (max 25 characters)    Provider number (max 9 digits)    Total fee for week (max $99,999.99) |

Figure 13.96.  Scenario for the Run Accounting Procedure use case.

Entity Class Modeling

Candidate entity classes are determined using noun extraction.

*Description of software product in a single paragraph:* The ChocAn system allows provid­ers to verify a person’s ChocAn membership status, and to submit a claim for a service provided to an active member.  The data required by the system can be main­tained by a ChocAn operator.  The ChocAn manager can request various reports.

.

*Identify the nouns/noun phrases:*The ChocAn system allows providers to verify a per­son’s ChocAn membership status, and to submit a claim for a service provided to an active member.  The data required by the system can be maintained by a ChocAn operator.  The ChocAn manager can request various reports.

 With regard to the nouns in the previous paragraph:

ChocAn System:  the software product itself

Provider: possible entity class

Person: can be a provider, member, operator or manager.

Membership status: possible attribute of member

Claim: possible entity class

Service: possible entity class

Data: describes all attributes of entity classes

ChocAn operator: actor, no information needs be stored about this actor

ChocAn manager: actor, no information needs be stored about this actor

The resulting entity-class diagram is shown in Figure 3.97.

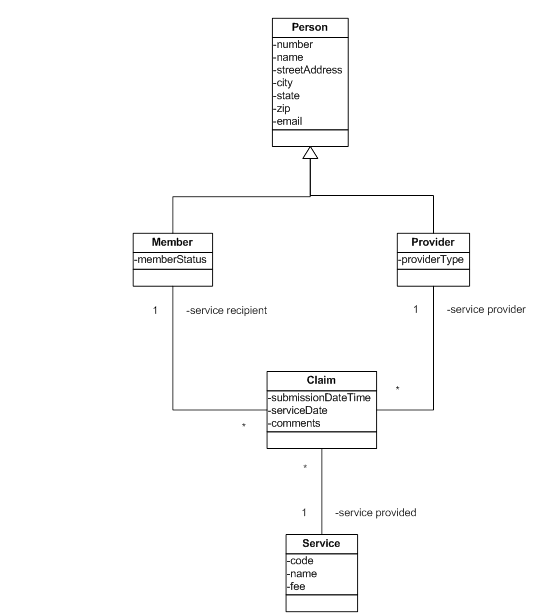


Figure 13.97.  The entity class diagram of the Chocoholics Anonymous product.

Control Classes:

**ChocAn System**

**ClaimSubmitter**

**Member Maintainer**

**Provider Maintainer**

**Service Maintainer**

**Provider Directory Generator**

**Provider Report Generator**

**Member Report Generator**

**Accounts Payable Report Generator**

**EFT Report Generator**

Boundary Classes:

**Provider Interface**

**Operator Interface**

**Manager Interface**

**Scheduler Interface**

**Provider Directory**

**Provider Report**

**Member Report**

**Accounts Payable Report**

**EFT Data Report**

Dynamic Modeling:

 Statecharts for the ChocAn product are shown in Figures 13.98(a), (b), (c), and (d).

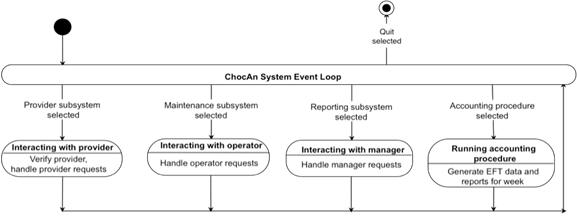


Figure 13.98(a).  The statechart for the Chocoholics Anonymous product.

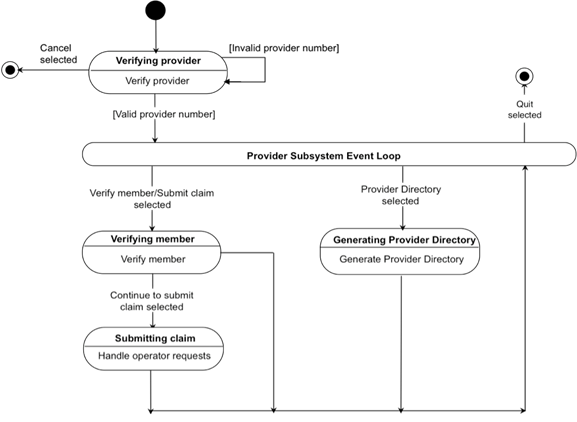


Figure 13.98(b).  Provider subsystem statechart.

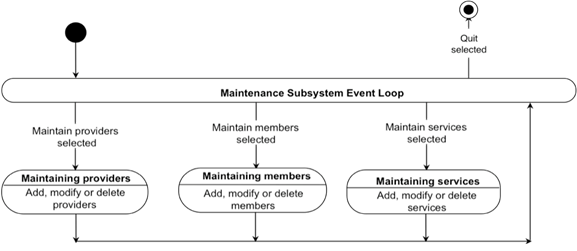


Figure 13.98(c).  Maintenance subsystem statechart.

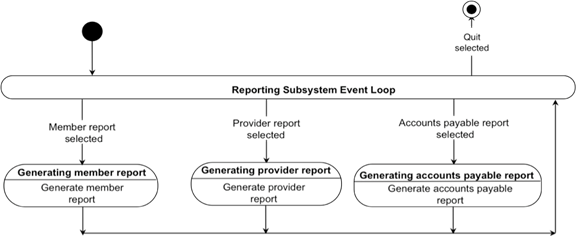


Figure 13.98(d).  Reporting subsystem statechart.

 We now realize the use cases.

Use caseManage Session:

 The class diagram is shown in Figure 13.99 (a).

 Consider the scenario of Figure 13.88.  The communication diagram is shown in Fig­ure 13.99 (b), the flow of events in Figure 13.99 (c) and the corresponding sequence dia­gram in Figure 13.99 (d).

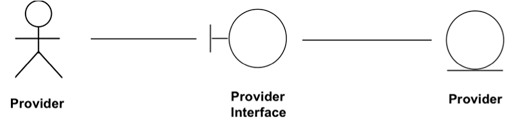


Figure 13.99 (a).  Class diagram showing the classes that realize the Manage Session use case of the ChocAn software product.

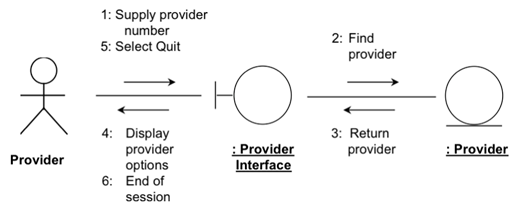


Figure 13.99 (b).  A communication diagram of the realization of the scenario of Figure 13.88 of the Manage Session use case of the ChocAn software product.

|  |
| --- |
| The provider supplies the necessary details. (1)  The software product finds the provider (2, 3), and displays the provider options (4).  The provider selects Quit (5) and the product ends the session (6). |

Figure 13.99 (c).  The flow of events of the realization of the scenario of Figure 13.88 of the Manage Session use case of the ChocAn software product.

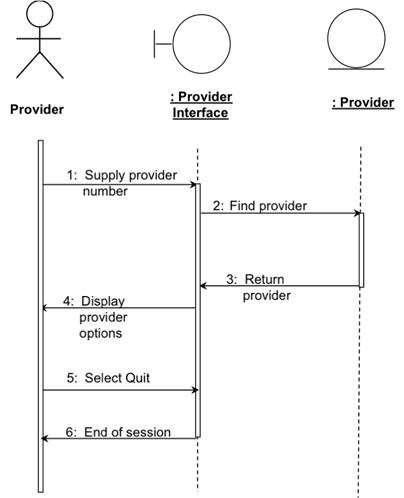


Figure 13.99 (d).  A sequence diagram equivalent to the communication diagram of Figure 13.99(b).  The flow of events therefore is shown in Figure 13.99(c).

Use caseVerify Member:

 The class diagram is shown in Figure 13.100(a).

 Consider the scenario of Figure 13.89.  The communication diagram is shown in Fig­ure 13.100(b).  The flow of events and the corresponding sequence dia­gram are shown in Figures 13.100(c) and 13.100(d), respectively.

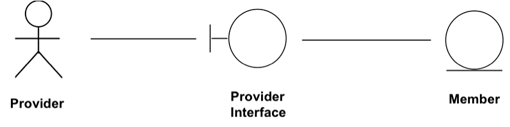


Figure 13.100(a).  Class diagram showing the classes that realize the Verify Member use case of the ChocAn software product.

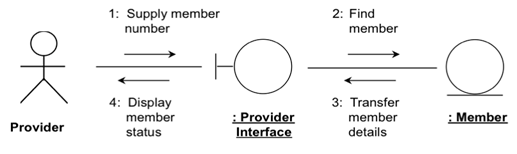


Figure 13.100(b).  A communication diagram of the realization of the scenario of Figure 13.89 of the Verify Member use case of the ChocAn software product.

|  |
| --- |
| The provider supplies the member number (1).  The software product finds the member details (2), and displays the member status (3–4). |

Figure 13.100(c).  The flow of events of the realization of the scenario of Figure 13.89 of the Verify Member use case of the ChocAn software product.

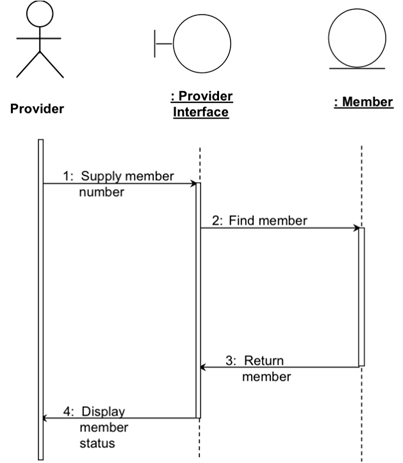


Figure 13.100(d).  A sequence diagram equivalent to the communication diagram of Figure 13.100(b).  The flow of events therefore is shown in Figure 13.100(c).

Use caseSubmit Claim:

 The class diagram is shown in Figure 13.101(a).

 Consider the scenario of Figure 13.90.  The communication diagram is shown in Fig­ure 13.101(b), the flow of events in Figure 13.101(c) and the corresponding sequence dia­gram in Figure 13.101(d).

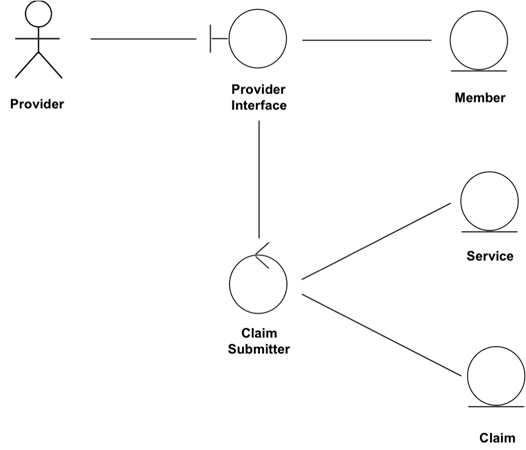


Figure 13.101(a).  Class diagram showing the classes that realize the Submit Claim use case of the ChocAn software product.

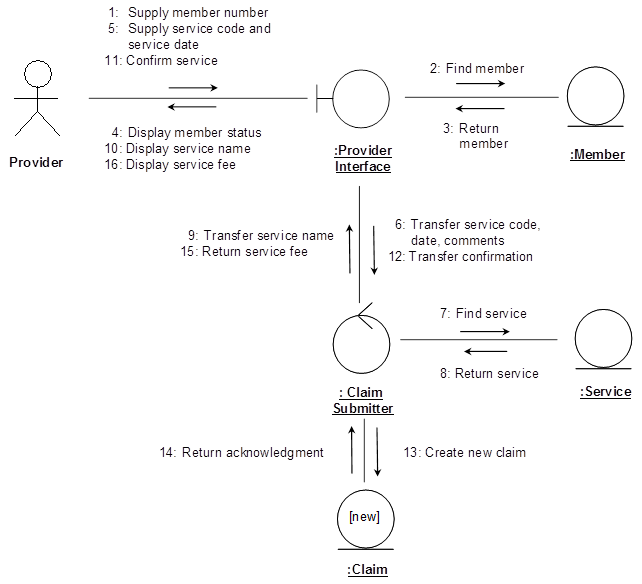


Figure 13.101(b).  A communication diagram of the realization of scenario of Figure 13.90 of the Submit Claim use case of the ChocAn software product.

|  |
| --- |
| The provider supplies the member number (1).  The software product finds and transfers the corresponding member (2–3), and displays the member’s status (4).  The provider supplies the service code and other data, which the product uses to find and return the service name to the provider (5–10).  The provider confirms the service (11-12), and the product creates a new claim and displays the service fee (13–16) |

Figure 13.101(c).  The flow of events of the realization of the scenario of Figure 13.90 of the Submit Claim use case of the ChocAn software product.

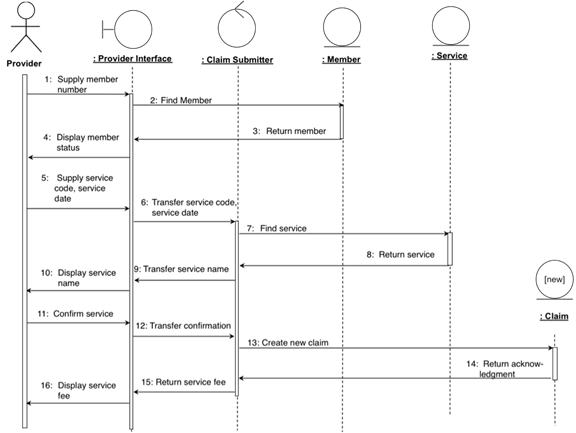


Figure 13.101(d).  A sequence diagram equivalent to the communication diagram of Figure 13.101(b).  The flow of events therefore is shown in Figure 13.101(c).

Use caseReceive Provider Directory:

 The class diagram is shown in Figure 13.102(a).

 Consider the scenario of Figure 13.91.  The communication diagram is shown in Fig­ure 13.102(b).  The flow of events and the corresponding sequence dia­gram are shown in Figures 13.102(c) and 13.102(d), respectively.

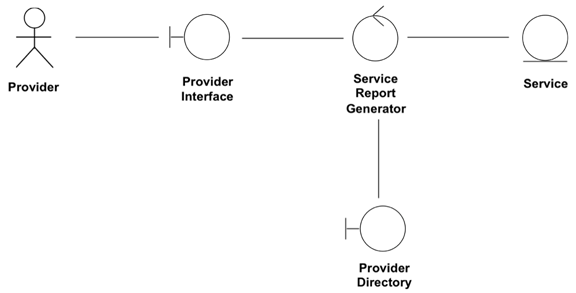


Figure 13.102(a).  Class diagram showing the classes that realize the Receive Provider Directory use case of the ChocAn software product.

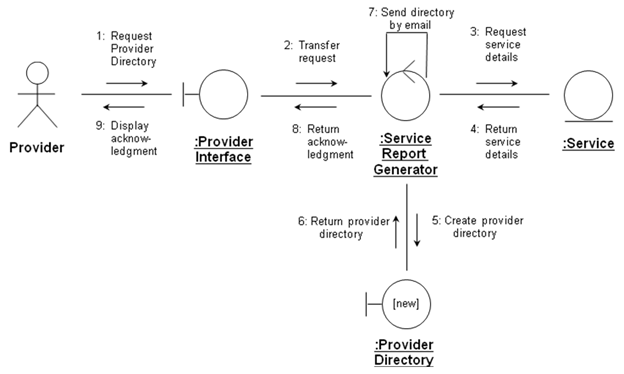


Figure 13.102(b).  A communication diagram of the realization of the scenario of Figure 13.91 of the Receive Provider Directory use case of the ChocAn software product.

|  |
| --- |
| The provider requests the provider directory (1).  The software product transfers the request (2).  The product obtains the service details (2–4), and uses this information to create the service directory (5, 6), which it then sends by e-mail (7).  It then displays an acknowledgment to the provider (8, 9). |

Figure 13.102(c).  The flow of events of the realization of the scenario of Figure 13.91 of the Receive Provider Directory use case of the ChocAn software product.

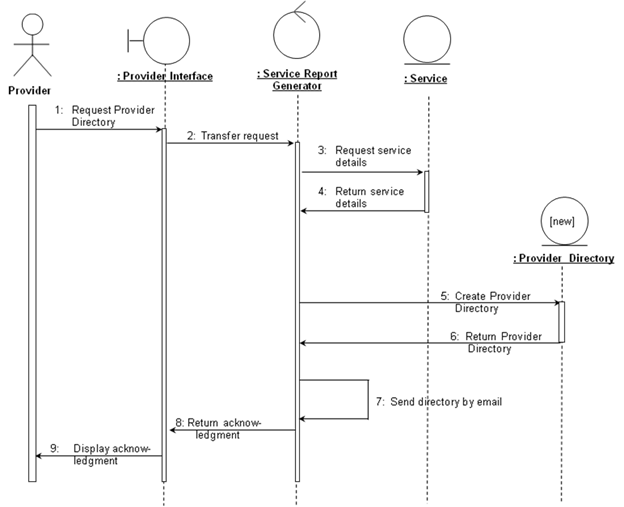


Figure 13.102(d).  A sequence diagram equivalent to the communication diagram of Figure 13.102(b).  The flow of events therefore is shown in Figure 13.102(c).

Use caseMaintain Member:

 The class diagram is shown in Figure 13.103(a).

 Consider the add a new member scenario of Figure 13.92(a).  The communication dia­gram is shown in Fig­ure 13.103(b), the flow of events in Figure 13.103(c) and the corre­sponding sequence dia­gram in Figure 13.103(d).

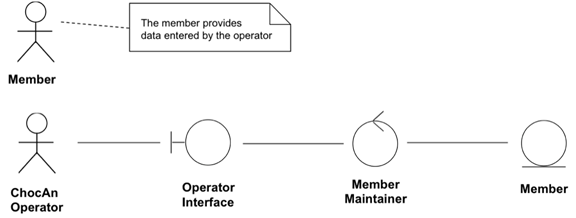


Figure 13.103(a).  Class diagram showing the classes that realize the Maintain Member use case of the ChocAn software product.

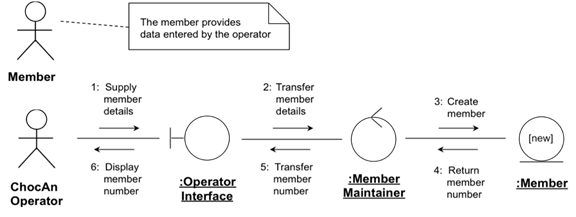


Figure 13.103(b).  A communication diagram of the realization of the add a member scenario of Figure 13.92(a) of the Maintain Member use case of the ChocAn software product.

|  |
| --- |
| The member supplies the necessary details, entered by the ChocAn operator (1).  The software product transfers the details, creates a new member record (2–3), and returns and displays the new member number (4–6). |

Figure 13.103(c).  The flow of events of the realization of the scenario of Figure 13.92(a) of the Maintain Member use case of the ChocAn software product.

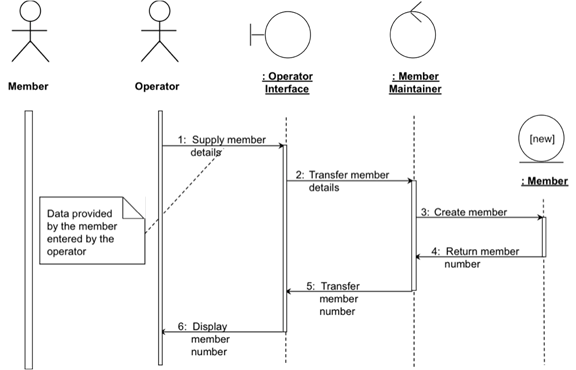


Figure 13.103(d).  A sequence diagram equivalent to the communication diagram of Figure 13.103(b).  The flow of events therefore is shown in Figure 13.103(c).

 Now consider the update a member scenario of the Maintain Member use case (Figure 13.92(b)).  The communication diagram is shown in Fig­ure 13.103(e), the flow of events in Figure 13.103(f) and the corresponding sequence dia­gram in Figure 13.103(g).

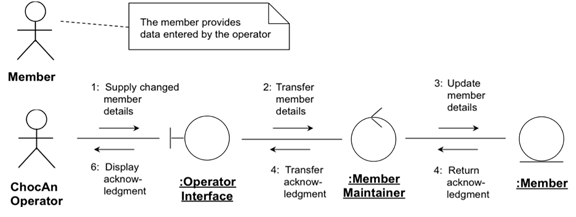


Figure 13.103(e).  A communication diagram of the realization of the update a member scenario of Figure 13.92(b) of the Maintain Member use case of the ChocAn software product.

|  |
| --- |
| The member supplies the changed details, entered by the ChocAn operator (1).  The software product transfers and updates the details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.103(f).  The flow of events of the realization of the scenario of Figure 13.92(b) of the Maintain Member use case of the ChocAn software product.

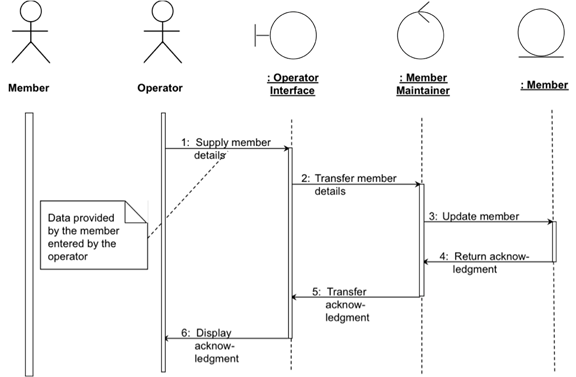


Figure 13.103(g).  A sequence diagram equivalent to the communication diagram of Figure 13.103(e).  The flow of events therefore is shown in Figure 13.103(f).

 Finally, consider the delete a member scenario of the Maintain Member use case (Fig­ure 13.92(c)).  The communication diagram is shown in Fig­ure 13.103(h), the flow of events in Figure 13.103(i) and the corresponding sequence dia­gram in Figure 13.103(j).

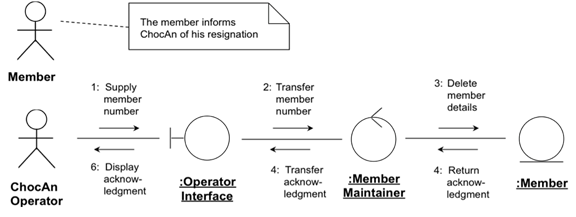


Figure 13.103(h).  A communication diagram of the realization of the delete a member scenario of Figure 13.92(c) of the Maintain Member use case of the ChocAn software product.

|  |
| --- |
| The member supplies his or her member number, entered by the ChocAn operator (1).  The software product transfers and deletes the member details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.103(i).  The flow of events of the realization of the scenario of Figure 13.92(c) of the Maintain Member use case of the ChocAn software product.

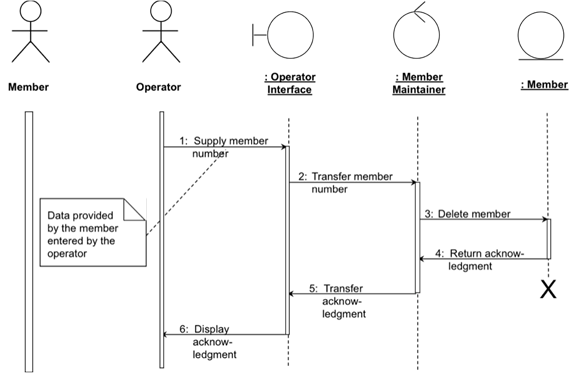


Figure 13.103(j).  A communication diagram of the realization of the delete a member scenario of Figure 13.92(c) of the Maintain Member use case of the ChocAn software product.

Use caseMaintain Provider:

 The class diagram is shown in Figure 13.104(a).

 Consider the add a new provider scenario of Figure 13.93(a).  The communication dia­gram is shown in Fig­ure 13.104(b), the flow of events in Figure 13.104(c) and the corre­sponding sequence dia­gram in Figure 13.104(d).

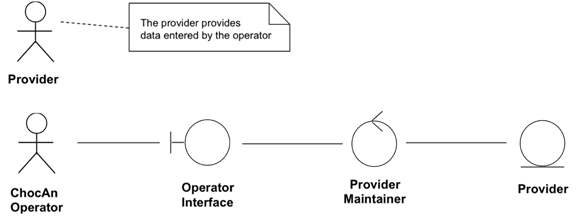


Figure 13.104(a).  Class diagram showing the classes that realize the Maintain Provider use case of the ChocAn software product.

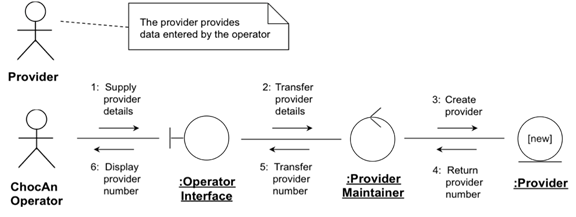


Figure 13.104(b).  A communication diagram of the realization of the add a provider scenario of Figure 13.93(a) of the Maintain Provider use case of the ChocAn software product.

|  |
| --- |
| The provider supplies the necessary details, entered by the ChocAn operator (1).  The software product transfers the details and creates a new provider record (2–3), and returns and displays the provider number (4–6). |

Figure 13.104(c).  The flow of events of the realization of the scenario of Figure 13.93(a) of the Maintain Provider use case of the ChocAn software product.

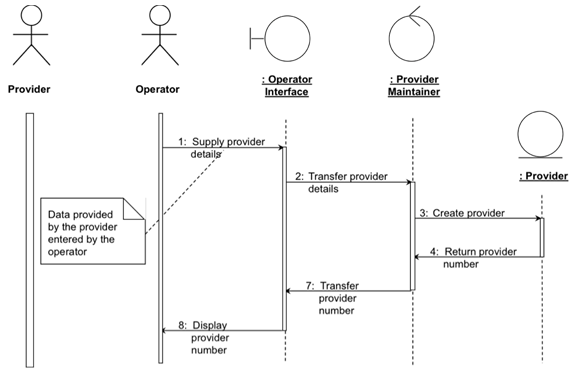


Figure 13.104(d).  A sequence diagram equivalent to the communication diagram of Figure 13.104(b).  The flow of events therefore is shown in Figure 13.104(c).

 Now consider the update a provider scenario of the Maintain Provider use case (Fig­ure 13.93(b)).  The communication diagram is shown in Fig­ure 13.104(e), the flow of events in Figure 13.104(f) and the corresponding sequence dia­gram in Figure 13.104(g).

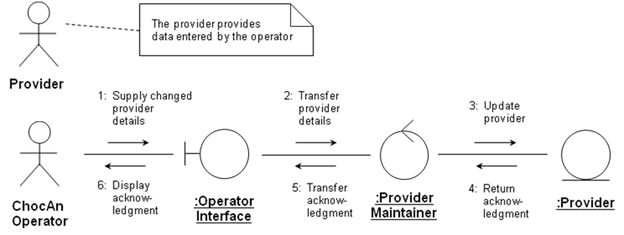


Figure 13.104(e).  A communication diagram of the realization of the update a provider scenario of Figure 13.93(b) of the Maintain Provider use case of the ChocAn software product.

|  |
| --- |
| The provider supplies the changed details, entered by the ChocAn operator (1).  The software product transfers and updates the details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.104(f).  The flow of events of the realization of the scenario of Figure 13.93(b) of the Maintain Provider use case of the ChocAn software product.

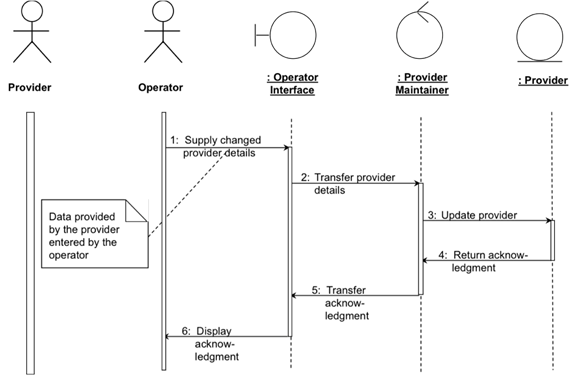


Figure 13.104(g).  A sequence diagram equivalent to the communication diagram of Figure 13.104(e).  The flow of events therefore is shown in Figure 13.104(f).

 Finally, consider the delete a provider scenario of the Maintain Provider use case (Figure 13.93(c)).  The communication diagram is shown in Fig­ure 13.104(h), the flow of events in Figure 13.104(i) and the corresponding sequence dia­gram in Figure 13.104(j).

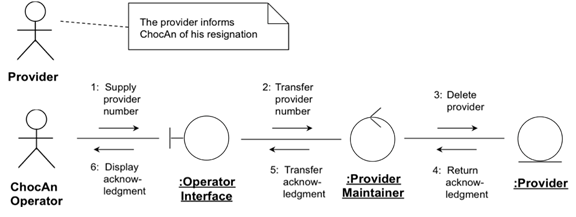


Figure 13.104(h).  A communication diagram of the realization of the delete a provider scenario of Figure 13.93(c) of the Maintain Provider use case of the ChocAn software product.

|  |
| --- |
| The provider supplies his or her provider number, entered by the ChocAn operator (1).  The software product transfers and deletes the provider details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.104(i).  The flow of events of the realization of the scenario of Figure 13.93(c) of the Maintain Provider use case of the ChocAn software product.

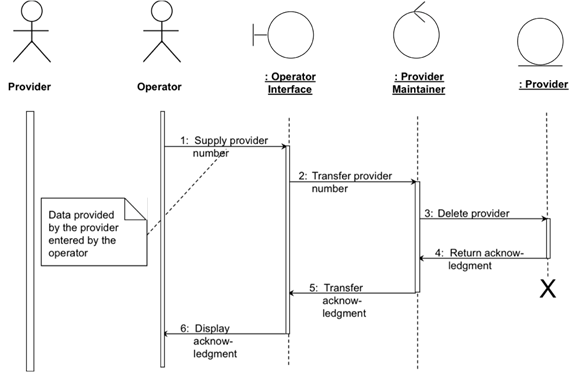


Figure 13.104(j).  A communication diagram of the realization of the delete a provider scenario of Figure 13.93(c) of the Maintain Provider use case of the ChocAn software product.

Use caseMaintain Service:

 The class diagram is shown in Figure 13.105(a).

 Consider the add a new service scenario of Figure 13.94(a).  The communication diagram is shown in Fig­ure 13.105(b), the flow of events in Figure 13.105(c) and the correspond­ing sequence dia­gram in Figure 13.105(d).

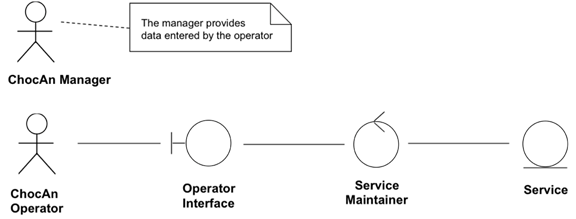


Figure 13.105(a).  Class diagram showing the classes that realize the Maintain Service use case of the ChocAn software product.

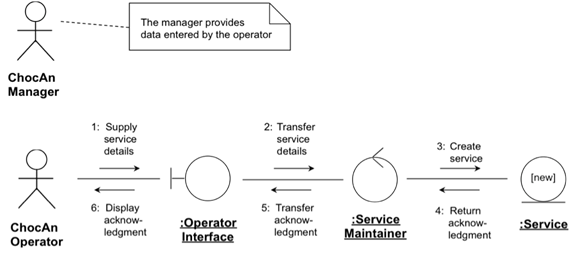


Figure 13.105(b).  A communication diagram of the realization of the add a service scenario of Figure 13.94(a) of the Maintain Service use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the necessary service details, entered by the ChocAn operator (1).  The software product transfers the details and creates a new service record (2–3), and sends an acknowledgment to the manager (4–6). |

Figure 13.105(c).  The flow of events of the realization of the scenario of Figure 13.94(a) of the Maintain Service use case of the ChocAn software product.

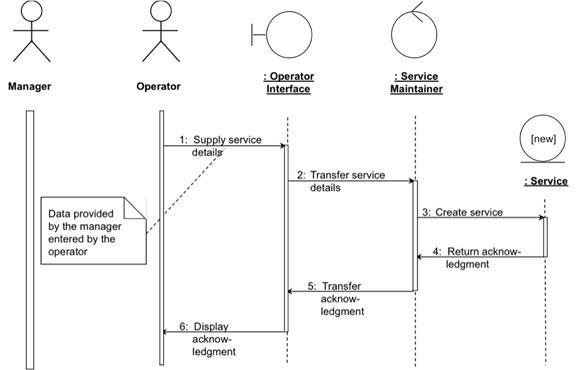


Figure 13.105(d).  A sequence diagram equivalent to the communication diagram of Figure 13.104(b).  The flow of events therefore is shown in Figure 13.104(c).

 Now consider the update a service scenario of the Maintain Service use case (Figure 13.94(b)).  The communication diagram is shown in Figure 13.105(e), the flow of events in Figure 13.105(f) and the corresponding sequence diagram in Figure 13.105(g).

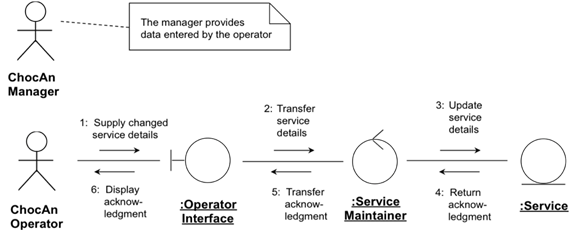


Figure 13.105(e).  A communication diagram of the realization of the update a service scenario of Figure 13.94(b) of the Maintain Service use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the changed service details, entered by the ChocAn operator (1).  The software product transfers and updates the details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.105(f).  The flow of events of the realization of the scenario of Figure 13.94(b) of the Maintain Service use case of the ChocAn software product.

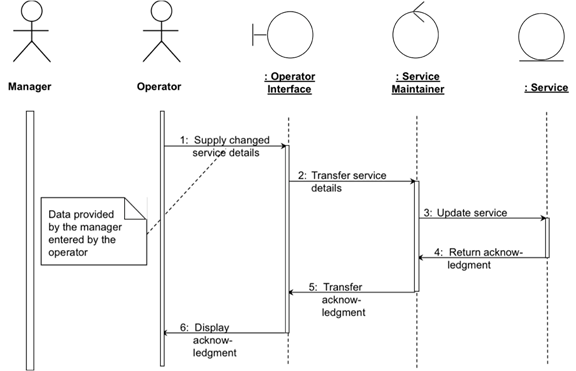


Figure 13.105(g).  A sequence diagram equivalent to the communication diagram of Figure 13.105(e).  The flow of events therefore is shown in Figure 13.105(f).

 Finally, consider the delete a service scenario of the Maintain Service use case (Fig­ure 13.94(c)).  The communication diagram is shown in Fig­ure 13.105(h), the flow of events in Figure 13.105(i) and the corresponding sequence dia­gram in Figure 13.105(j).

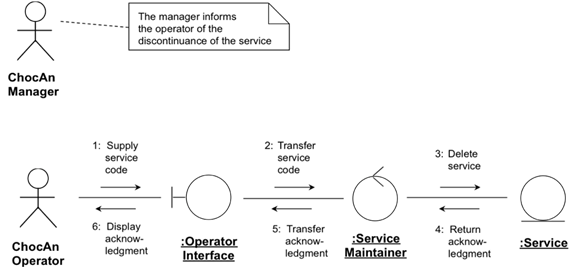


Figure 13.105(h).  A communication diagram of the realization of the delete a service scenario of Figure 13.94(c) of the Maintain Service use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the service code, entered by the ChocAn operator (1).  The software product transfers and deletes the service details (2–3), and returns and displays an acknowledgment (4–6). |

Figure 13.105(i).  The flow of events of the realization of the scenario of Figure 13.94(c) of the Maintain Service use case of the ChocAn software product.

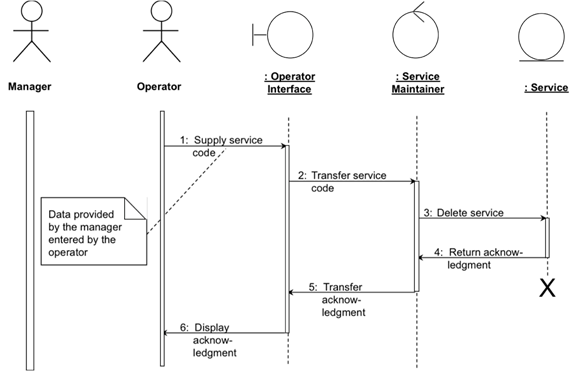


Figure 13.105(j).  A sequence diagram equivalent to the communication diagram of Figure 13.105(h).  The flow of events therefore is shown in Figure 13.105(i).

Use caseRequest Report:

 The class diagram is shown in Figure 13.106(a).

 Consider the provider report scenario of Figure 13.95(a).  The communication diagram is shown in Fig­ure 13.106(b), the flow of events in Figure 13.106(c) and the corresponding sequence dia­gram in Figure 13.106(d).

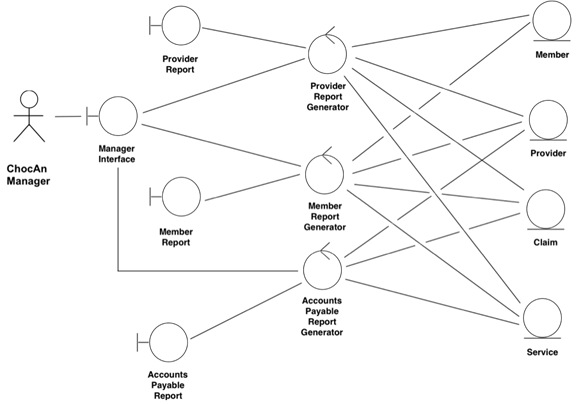


Figure 13.106(a).  Class diagram showing the classes that realize the Request Report use case of the ChocAn software product.

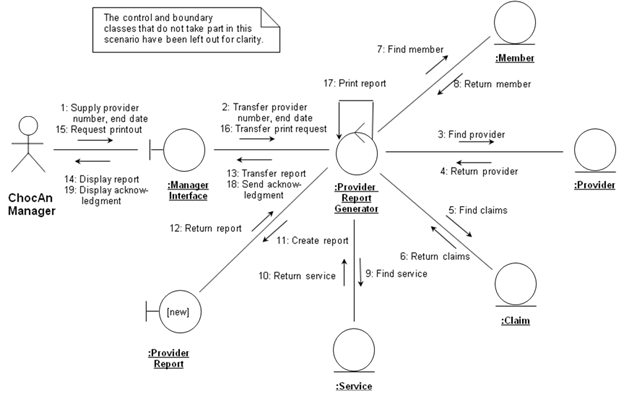


Figure 13.106(b).  A communication diagram of the realization of the provider report scenario of Figure 13.95(a) of the Request Report use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the relevant provider number and end date (1).  The software product transfers the details, finds and returns the provider (3–4), then finds and returns the relevant claims (5–6).  It finds and returns each member who has received services (7–8), and the relevant services (9–10).  The software product then creates the report and displays it to the manager (11–14).  The manager then requests a printout, which is performed (15–17), and an acknowledgment displayed (18–19). |

Figure 13.106(c).  The flow of events of the realization of the provider report scenario of Figure 13.95(a) of the Request Report use case of the ChocAn software product.

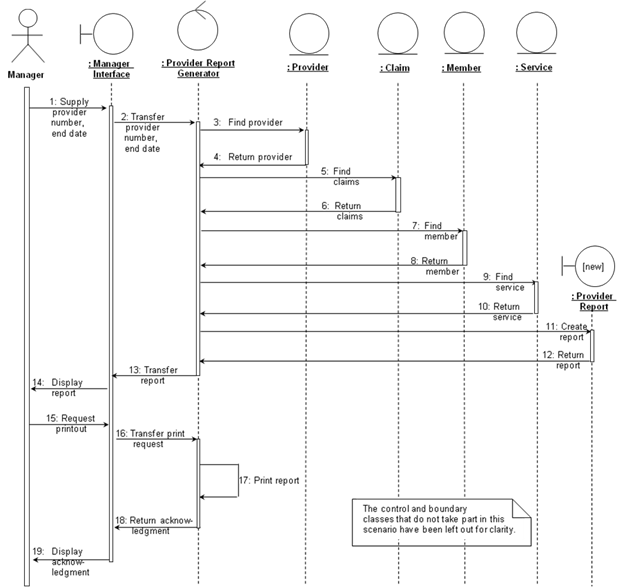


Figure 13.106(d).  A sequence diagram equivalent to the communication diagram of Figure 13.106(b).  The flow of events therefore is shown in Figure 13.106(c).

 Now consider the member report scenario of the Request Report use case (Figure 13.95(b)).  The communication diagram is shown in Fig­ure 13.106(e), the flow of events in Figure 13.106(f) and the corresponding sequence dia­gram in Figure 13.106(g).

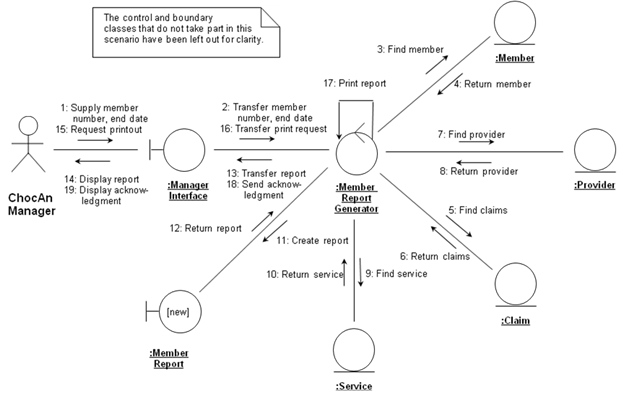


Figure 13.106(e).  A communication diagram of the realization of the member report scenario of Figure 13.95(b) of the Request Report use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the relevant member number and end date (1).  The software product transfers the details, finds and returns the member (3–4), then finds and returns the relevant claims (5–6).  It finds and returns each provider who has provided services to that member (7–8), and the relevant services (9–10).  The software product then creates the report and displays it to the manager (11–14).  The manager then requests a printout, which is performed (15–17), and an acknowledgment displayed (18–19). |

Figure 13.106(f).  The flow of events of the realization of the scenario of Figure 13.95(b) of the Request Report use case of the ChocAn software product.

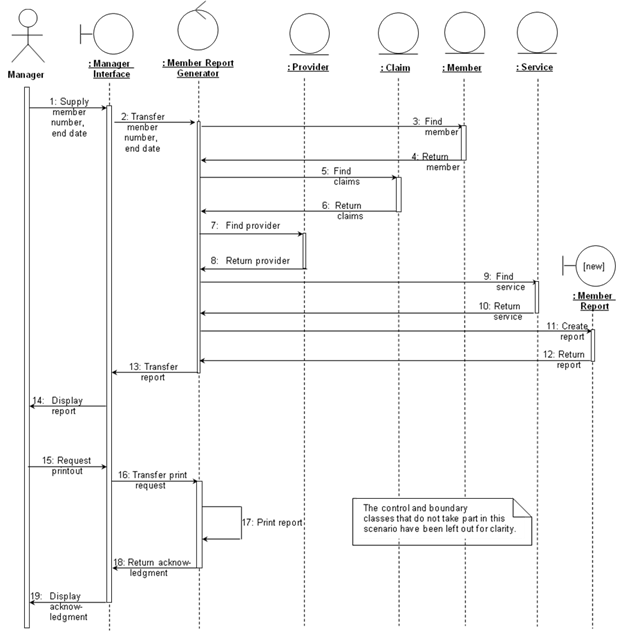


Figure 13.106(g).  A sequence diagram equivalent to the communication diagram of Figure 13.106(e).  The flow of events therefore is shown in Figure 13.106(f).

 Finally, consider the accounts payable scenario of the Request Report use case (Fig­ure 13.95(c)).  The communication diagram is shown in Fig­ure 13.106(h), the flow of events in Figure 13.106(i) and the corresponding sequence dia­gram in Figure 13.106(j).

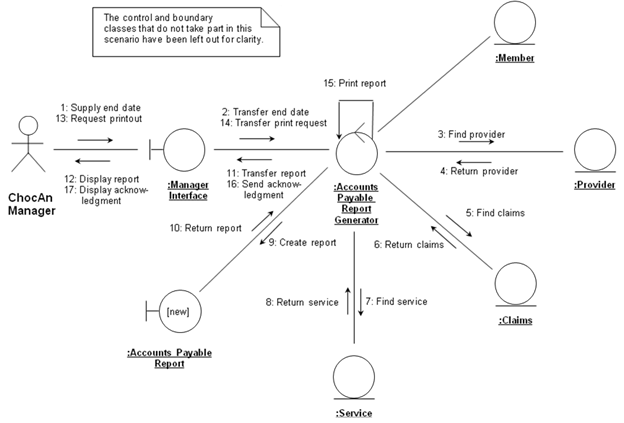


Figure 13.106(h).  A communication diagram of the realization of the accounts payable scenario of Figure 13.95(c) of the Request Report use case of the ChocAn software product.

|  |
| --- |
| The manager supplies the end date (1).  The software product transfers the end date, finds and returns the providers who provided services (3–4), then finds and returns the relevant claims (5–6).  It finds and returns the services provided by each provider (7–8).  The software product then creates the report and displays it to the manager (9–12).  The manager then requests a printout, which is performed (13–15), and an acknowledgment displayed (16–17). |

Figure 13.106(i).  The flow of events of the realization of the scenario of Figure 13.95(c) of the Request Report use case of the ChocAn software product.

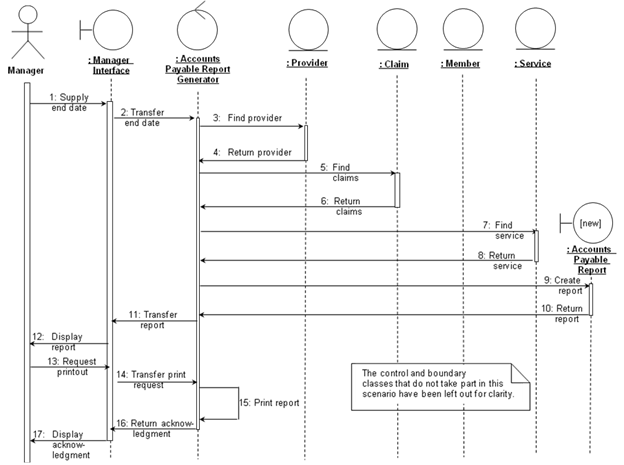


Figure 13.106(j).  A sequence diagram equivalent to the communication diagram of Figure 13.106(h).  The flow of events therefore is shown in Figure 13.106(i).

Use caseRun Accounting Procedure:

 The class diagram is shown in Figure 13.107(a).

 Consider the EFT report scenario of Figure 13.96.  The communication diagram is shown in Fig­ure 13.107(b), the flow of events in Figure 13.107(c) and the corresponding sequence dia­gram in Figure 13.107(d).

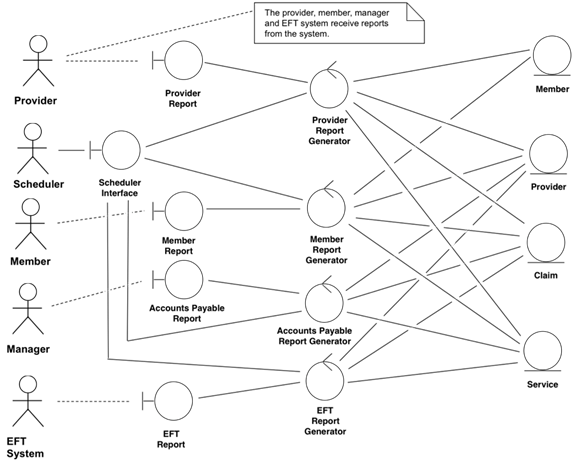


Figure 13.107(a).  Class diagram showing the classes that realize the Run Accounting Procedure use case of the ChocAn software product.

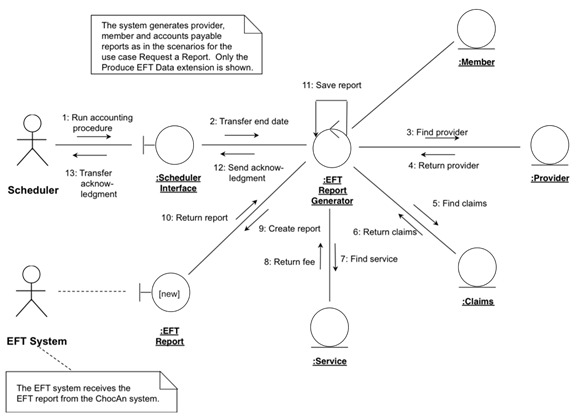


Figure 13.107(b).  A communication diagram of the realization of EFT report scenario of Figure 13.96 of the Run Accounting Procedure use case of the ChocAn software product.

|  |
| --- |
| The scheduler runs the accounting procedure (1).  The software product transfers the end date, finds and returns each provider (2–4), then finds and returns the relevant claims (5–6).  It finds each service and returns the corresponding fee (7–8).  The software product then creates and saves the report (9–11) and sends an acknowledgment to the scheduler (12–13). |

Figure 13.107(c).  The flow of events of the realization of the scenario of Figure 13.96 of the Run Accounting Procedure use case of the ChocAn software product.

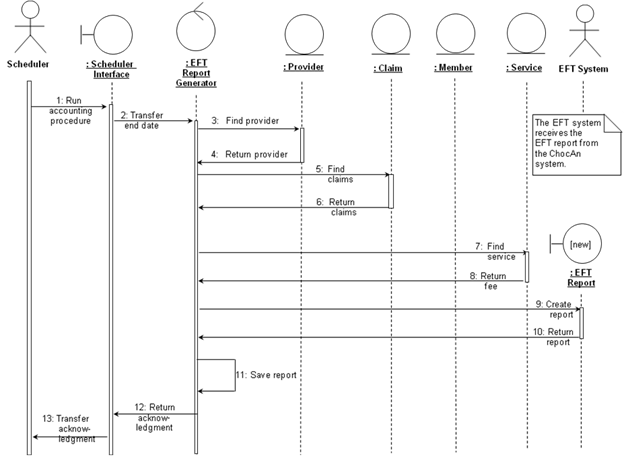


Figure 13.107(d).  A sequence diagram equivalent to the communication diagram of Figure 13.107(b).  The flow of events therefore is shown in Figure 13.107(c).

Increment the class diagram.

 In the course of realizing the various use cases, interrelationships between many of the classes became apparent; these interrelationships are reflected in the class diagrams of Figures 13.99(a) through 13.107(a).

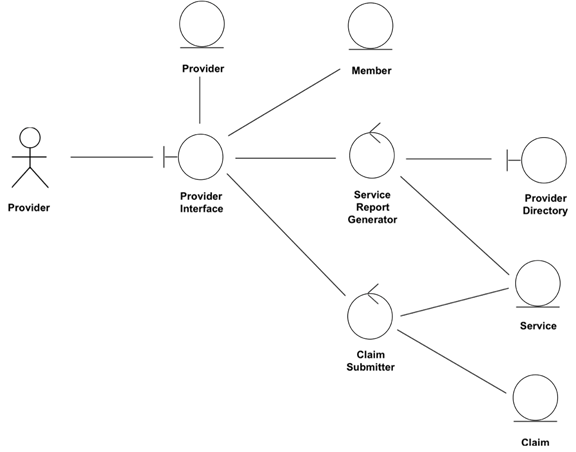


Figure 13.108(a).  A class diagram combining the class diagrams of Figures 13.99(a) through 13.102(a)

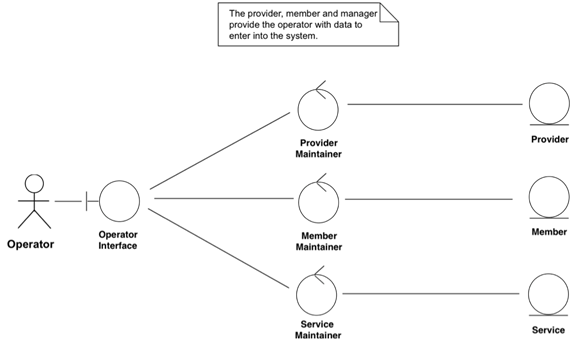


Figure 13.108(b).  A class diagram combining the class diagrams of Figures 13.103(a) through 13.105(a).

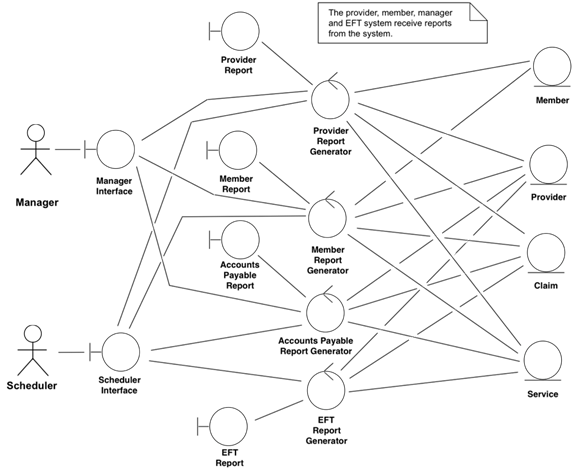


Figure 13.108(c).  A class diagram combining the class diagrams of Figures 13.106(a) through 13.107(a).

 This concludes the analysis workflow.

[Comment](javascript:;)