

# Contents

[LOGON] USE CASE – UC001	4
1. Brief Description	4
2. Actors	4
3. Pre-Conditions	4
4. Basic Flow	4
5. Alternate/Exception Flows	5
6. Post Conditions	5
7. Supplemental Requirements	5
8. Visual Model	6
[LOGOUT] USE CASE – UC002	7
9. Brief Description	7
10. Actors	7
11. Pre-Conditions	7
12. Basic Flow	7
13. Alternate/Exception Flows	8
14. Post Conditions	8
15. Supplemental Requirements	8
16. Visual Model	8
[ADD] USE CASE – UC003	10
17. Brief Description	10
18. Actors	10
19. Pre-Conditions	10
20. Basic Flow	11
21. Alternate/Exception Flows	11
22. Post Conditions	11
23. User logged out	12
24. Application terminated	12
25. Supplemental Requirements	12
26. Visual Model	12
[MODIFY] USE CASE – UC004	13
27. Brief Description	13

28.	Actors	13
29.	Pre-Conditions	13
30.	Basic Flow	14
31.	Alternate/Exception Flows	14
32.	Post Conditions	15
33.	User logged out	15
34.	Application terminated	15
35.	Supplemental Requirements	15
36.	Visual Model	15
[DISPLAY] USE CASE – UC005		17
37.	Brief Description	17
38.	Actors	17
39.	Pre-Conditions	17
40.	Basic Flow	18
41.	Alternate/Exception Flows	18
42.	Post Conditions	18
43.	User logged out	19
44.	Application terminated	19
45.	Supplemental Requirements	19
46.	Visual Model	19
[DELETE] USE CASE – UC006		21
47.	Brief Description	21
48.	Actors	21
49.	Pre-Conditions	21
50.	Basic Flow	21
51.	Alternate/Exception Flows	22
52.	Post Conditions	22
53.	User logged out	23
54.	Application terminated	23
55.	Supplemental Requirements	23
56.	Visual Model	23

Document Title	<i>Use cases for diary activities application</i>
Document Owner	Yasir Satti
Version	1.0
Status	
Date	04/03/2020

# [LOGON] USE CASE – UC001

---

## [LOGON] USE CASE – UC001

### 1. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to logon to the application and it ends the logon is successful or being terminated.

### 2. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 3. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

1. A username and password must exists in the system before logon attempt
2. Internet connection is working

### 4. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

1. User types application url into a web browser
2. After the page loads it shows the logon screen requesting username and password
3. User enter username and password and presses enter
4. Application landing page comes up

## 5. **ALTERNATE/EXCEPTION FLOWS**

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it's addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, "Use Case Ends."*

3a – If username is incorrect then ask for correct username.

3b - Terminate after 3 wrong attempts

3c - If password is not correct then ask for correct password.

3d - Terminate after 3 wrong attempts

## 6. **POST CONDITIONS**

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

3b & 3d – application logon terminated

4a – user logged on the application and can see the landing page

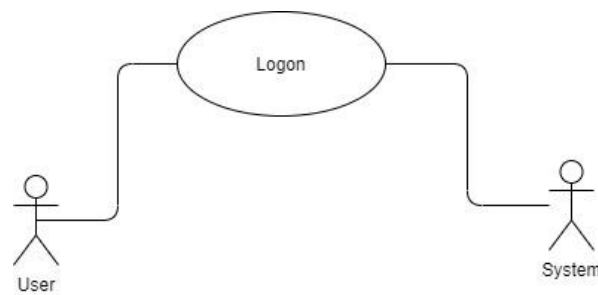
## 7. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've*

*also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

## 8. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*



# [LOGOUT] USE CASE – UC002

---

## [LOGOUT] USE CASE – UC002

### 9. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to logout from the application and ends when the user is logged out.

### 10. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 11. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

1. A username and password must exist in the system before logon attempt
2. Internet connection is working

### 12. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

5. User selects the sign-off button
6. Application terminates

### 13. **ALTERNATE/EXCEPTION FLOWS**

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it's addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, "Use Case Ends."*

N/A

### 14. **POST CONDITIONS**

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

2a – user logged out

2b – application terminated

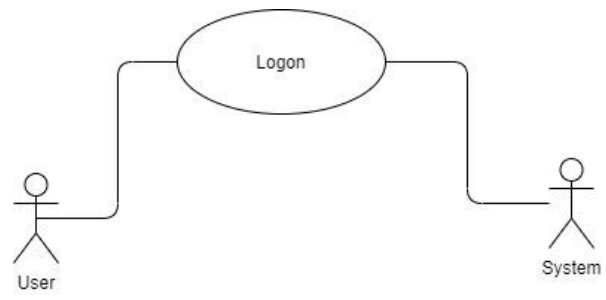
### 15. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

### 16. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*





# [ADD] USE CASE – UC003

---

## [Add] USE CASE – UC003

Document Title	Logout to diary activities application
Document Owner	Yasir Satti
Version	1.0
Status	
Date	04/03/2020

### 17. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to add an activity, then enter activity details and select to add them.

### 18. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 19. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

3. User logged on
4. Internet connection is working

## 20. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

7. The user selects “Add activity” button
8. The user enters:
  - a. Types Activity description
  - b. Selects Date
  - c. Selects Start time
  - d. Selects End time
  - e. Selects Achievement rating (scale: none is 0, 10 is completed)
  - f. Selects Joy rating (scale: did not like it at all is 0, 10 is being ecstatic)
9. The user selects “Add” button
10. The user is returned to landing page

## 21. ALTERNATE/EXCEPTION FLOWS

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it’s addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, “Use Case Ends.”*

N/A

## 22. POST CONDITIONS

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

New activity details added to the activity list

## 23. **USER LOGGED OUT**

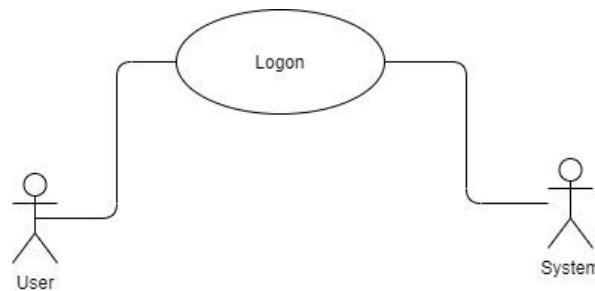
## 24. **APPLICATION TERMINATED**

## 25. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

## 26. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*



# [MODIFY] USE CASE – UC004

---

## [MODIFY] USE CASE – UC004

Document Title	Logout to diary activities application
Document Owner	Yasir Satti
Version	1.0
Status	
Date	04/03/2020

### 27. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to update or change details of an activity, then it ends when the user selects to save the new changes.

### 28. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 29. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

5. User logged on
6. Internet connection is working
7. The activity the user wants to update exists in the list

### 30. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

11. The user selects “Modify activity” button
12. The application prompts the user to select the activity date
13. The user selects the activity date
14. The user selects “Find” button
15. The application lists the activity
16. The user selects an activity for modification and presses “Edit”
17. The application displays the selected activity for editing
18. The user modifies any of the following fields:
  - a. Activity description
  - b. Date
  - c. Start time
  - d. End time
  - e. Achievement rating
  - f. Joy rating
19. The user selects “Add” button
20. The user is returned to landing page

### 31. ALTERNATE/EXCEPTION FLOWS

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it’s addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, “Use Case Ends.”*

N/A

## 32. POST CONDITIONS

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

New activity details added to the activity list

## 33. USER LOGGED OUT

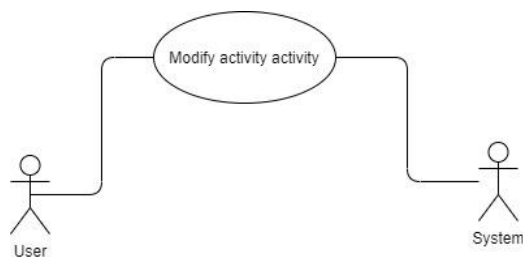
## 34. APPLICATION TERMINATED

## 35. SUPPLEMENTAL REQUIREMENTS

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

## 36. VISUAL MODEL

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*



# [DISPLAY] USE CASE – UC005

---

## [DISPLAY] USE CASE – UC005

Document Title	Logout to diary activities application
Document Owner	Yasir Satti
Version	1.0
Status	
Date	04/03/2020

### 37. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to display the list of activities, then it ends when the user selects close.

### 38. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 39. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

8. User logged on
9. Internet connection is working



#### 40. **BASIC FLOW**

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

21. The user selects “Display Activities” button
22. The application prompts the user to select the activity date
23. The user selects the activity date
24. The user selects “Find” button
25. The application lists the activities on the selected date
26. The user selects “Close”
27. The application returns user to landing page

#### 41. **ALTERNATE/EXCEPTION FLOWS**

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it’s addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, “Use Case Ends.”*

5a – if there are no activities on selected date then:

1. inform the user
2. prompt the user to select another date or cancel.
3. If user cancels, then return user to landing page

#### 42. **POST CONDITIONS**

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

N/A

#### 43. **USER LOGGED OUT**

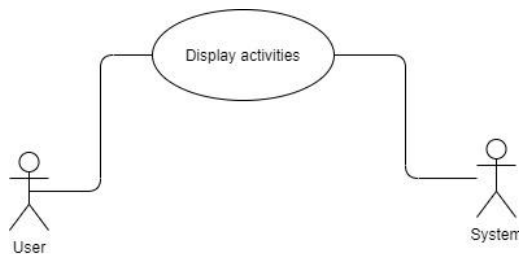
#### 44. **APPLICATION TERMINATED**

#### 45. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

#### 46. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*



Text

# [DELETE] USE CASE – UC006

---

## [DELETE] USE CASE – UC006

### 47. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to display the list of activities, then it ends when the user selects close.

### 48. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 49. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

10. User logged on
11. Internet connection is working

### 50. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

28. The user selects “Delete an Activity” button
29. The application prompts the user to select the activity date
30. The user selects the activity date
31. The user selects “Find” button

32. The application lists the activities on the selected date
33. The user selects an activity to delete
34. The user presses "Delete" button
35. The application deletes the selected activity from the list
36. The application returns user to landing page

## 51. **ALTERNATE/EXCEPTION FLOWS**

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it's addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, "Use Case Ends."*

5a – if there are no activities on selected date then:

4. inform the user
5. prompt the user to select another date or cancel.
6. If user cancels, then return user to landing page

## 52. **POST CONDITIONS**

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

N/A

### 53. **USER LOGGED OUT**

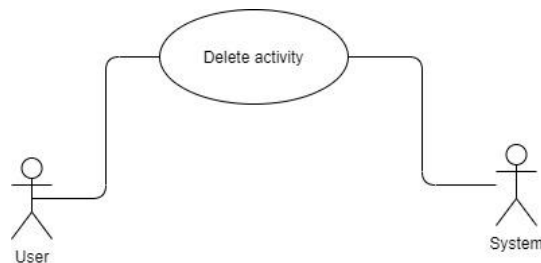
### 54. **APPLICATION TERMINATED**

### 55. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

### 56. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*



# [CREATE] USE CASE – UC007

---

## [Create] USE CASE – UC007

Document Title	Logout to diary activities application
Document Owner	Yasir Satti
Version	1.0
Status	
Date	12/03/2020

### 57. BRIEF DESCRIPTION

*Insert a 1-2 sentence description of this use case. Be sure to include a starts when / ends when statement to clarify the beginning and ending points of the scope of this process or piece of functionality.*

This use case starts when the user decides to create a new activity to add to a list where the user can select from this list instead of typing details each time the user wants to add an activity, then enter activity details and selects to create it.

### 58. ACTORS

*List any roles or systems involved with this process or use case. A person or system fulfilling a role will be the actor in one of the steps.*

- Professional
- System

### 59. PRE-CONDITIONS

*List anything that must be true before this process or functionality begins. Preconditions should be states that a system can validate to be true. A common example is that a specific Actor has logged into the System.*

1. User logged on
2. Internet connection is working

### 60. BASIC FLOW

*The basic flow is the normal course of events, otherwise called the “happy path.” Ask yourself, what happens most of the time and you’ll discover the steps that belong here. You’ll want your basic flow to cover the full scope of activities between the starts when and ends when.*

*Create a numbered list of each step below. I recommend using the Word “numbered list” functionality to automatically number the list.*

37. The user selects “Add activity” button
38. The user enters:
  - a. Types Activity description
  - b. Selects Date
  - c. Selects Start time
  - d. Selects End time
  - e. Selects Achievement rating (scale: none is 0, 10 is completed)
  - f. Selects Joy rating (scale: did not like it at all is 0, 10 is being ecstatic)
39. The user selects “Add” button
40. The user is returned to landing page

## 61. **ALTERNATE/EXCEPTION FLOWS**

*An alternate flow is a variation from the basic flow. Alternatives can be triggered at any step in the basic flow and often reinsert the actors back into the basic flow.*

*An exception flow is an error, or a negative condition. When an exception is encountered, it prevents the process from finishing through to its conclusion until it’s addressed.*

*Number your alternate and exception flows to indicate the step at which the variation occurs. For example, a variation on step 3 could be listed as 3a and a second variation as 3b, and so forth.*

*Describe the alternate functionality and then identify at what step in the basic flow this variation picks back up. For exception flows that result in the use case ending, simply write, “Use Case Ends.”*

N/A

## 62. **POST CONDITIONS**

*Post-conditions indicate what must be true of the state of the system after the steps of the use case are complete. These should be true for the basic flow and all alternate flows. Exception flows may have different post-conditions or none at all.*

New activity details added to the activity list

### 63. **USER LOGGED OUT**

### 64. **APPLICATION TERMINATED**

### 65. **SUPPLEMENTAL REQUIREMENTS**

*This is a special section I use to hold miscellaneous requirements related to the use case. Often you'll find BAs including a Business Rules section or other collection of information related to the use case. These may or may not be actual requirements – you'll want to establish a clear pattern and communicate that clearly and ensure it's consistent with how your organization documents this type of requirement. I've also used this section to capture the most salient decisions and notes so they are stored right with the use case for future consideration.*

### 66. **VISUAL MODEL**

*Many use cases are enhanced by a visual model. A simple work-flow diagram can be used to visually show the sequence of steps and alternate and exception flows. A user interface mock-up can be used to show a possible representation of these user requirements in an interface (or a desired representation). In some organizations, a more formal UML diagram may be appropriate.*

