**Use Cases:**

Now that there is a list of priorities that has gone through both functional and high-level technical server review, some work should be done to define certain key processes that the organization requires. These vary based on the output of the work. **Plan carefully for the items** that make sense for the business. A use case is a goal-focused set of processes that occur between users of a system and system itself. They can vary in depth depending on the level of complexity and established rigor for each organization. An example of use case might be the process of a project manager adding resource to a plan in Project Professional. It might look something like the high-level.

There are some common items that most organizations should map with a use case for Project Server. The Time and Task entries are great examples. There are many ways to use, and therefore configure, the system for these functions. Some questions to ask include the following:

1. Is it important to separate Time entry and approvals from project updates so that the resource of functional manager approves the time, and the project manager approves task updates and applies them to the plan?

2. **Must all time be recorded in the system, even non-project time**?

3. What types of non-project time should be recorded and how should that work?

4. Should vacation time planning be supported?

5. Should sick time require review and approval?

6. Do billable and nonbillable hours both have to be included?

7. Are all employees exempt from overtime, are they all hourly, or is the workforce a combination of both?

8. How should support time be handled?

9. How should maintenance time be handled?

10. Should time and task information be submitted weekly, monthly, or daily?

**Use cases should be written for each time and task type, such as the following:**

\*Regular project

\* Administrative time – vacation, sick, company events, and so on

\* Support-planned

\*Support-incident based

**Other related use cases might include such items as the following:**

**\*General use cases**

\*Propose a potential project

\*Review, approve, and reject a proposed project

\*View the status of a project, program, or portfolio

\*View the status of issues, risks, or deliverables

\*View and print reports

\***Project manager-focused use case**

\*Accept and initiate a new or approved project or program

\*View resource availability and assignments

\*Request resources

\*Build project team (if permission allow)

\*Assignment of resources to tasks

\*Update project status (Progress updates and plan maintenance)

\*Create and assign issues, risks, or deliverables

**\*Use case for resource manager**

\*Add/update resource data

\*Review, approve, and reject timesheets

\*Allocate resources to a project

\* Approve administrative time

**\*Team Member**

\*View and update task progress

\*Create, update, and submit a timesheet

\*Create or update issues, risks, or deliverables

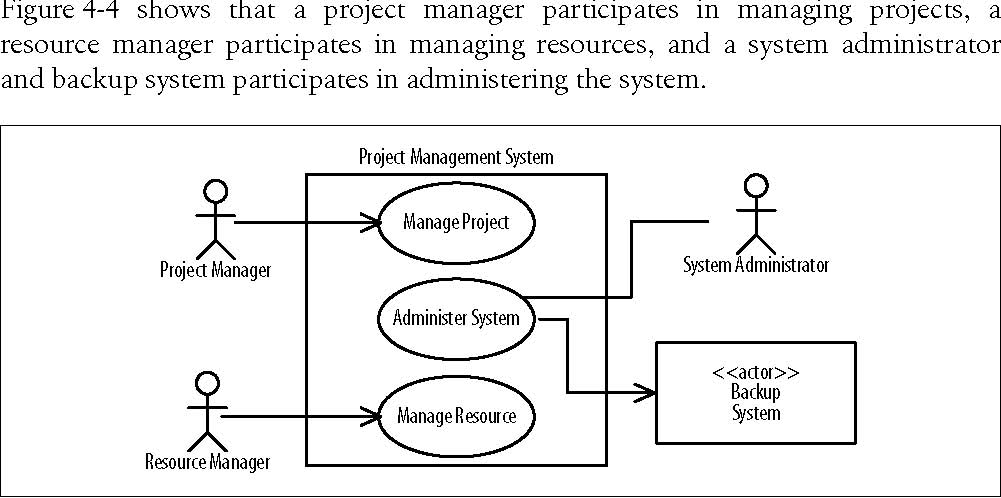
\*Request non project/admin time with and without approval option enabled

The use cases should include variable paths, often called scenarios that could occur within each of them. Most important, the goal in the use case must have a measurable or provable component. The next

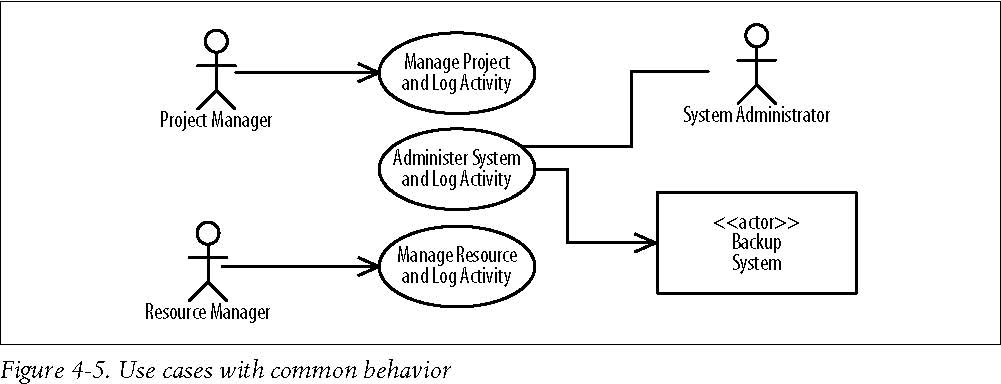
A use case is a functional requirement that is described from the perspective of the users of a system. For example, functional requirements for the project management system include: security functionality (such as allowing users to log in and out of the system), inputting of data, processing of data, generation of reports, and so forth

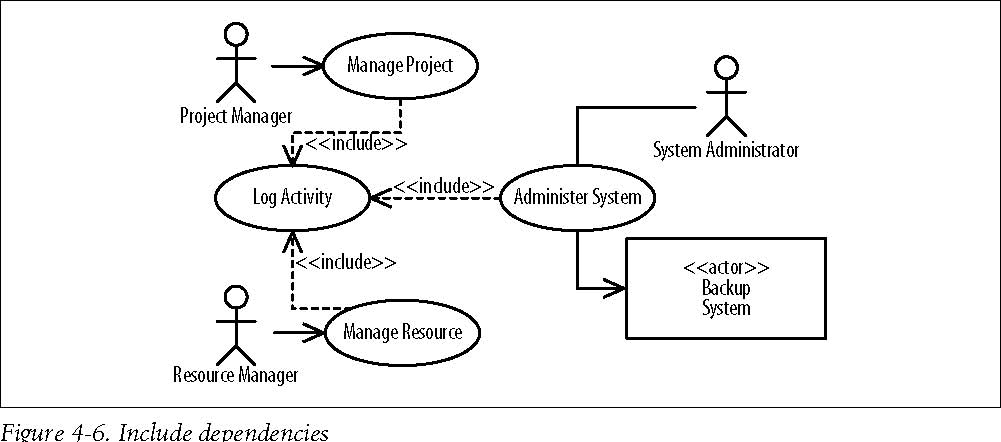
|  |  |
| --- | --- |
| Project Management System   |  | | --- | |  | |

**Communication Associates:-**

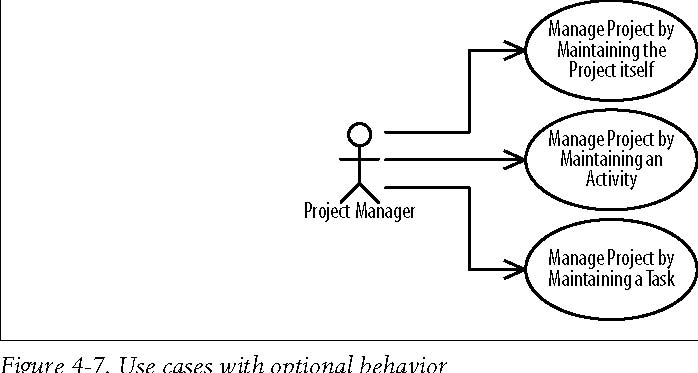


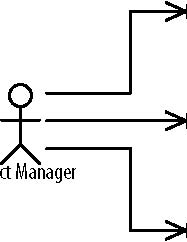
**Include Dependencies**

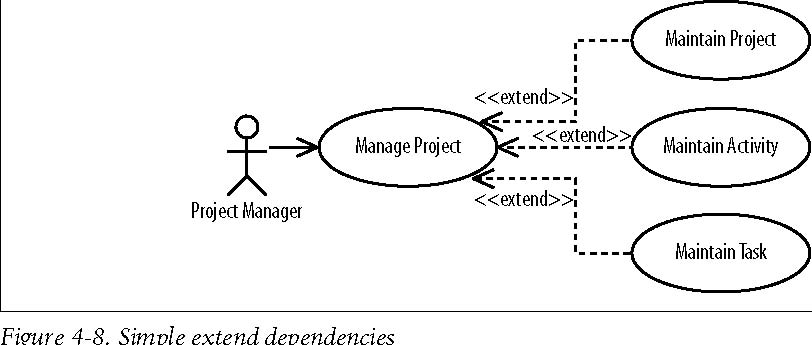


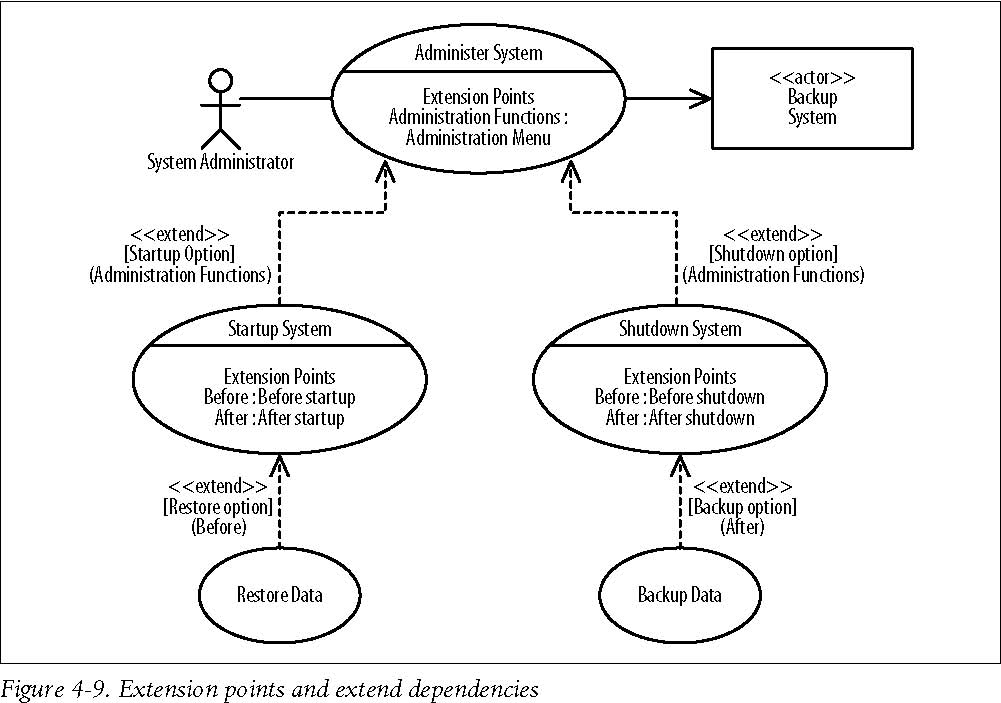


**Extend Dependencies**

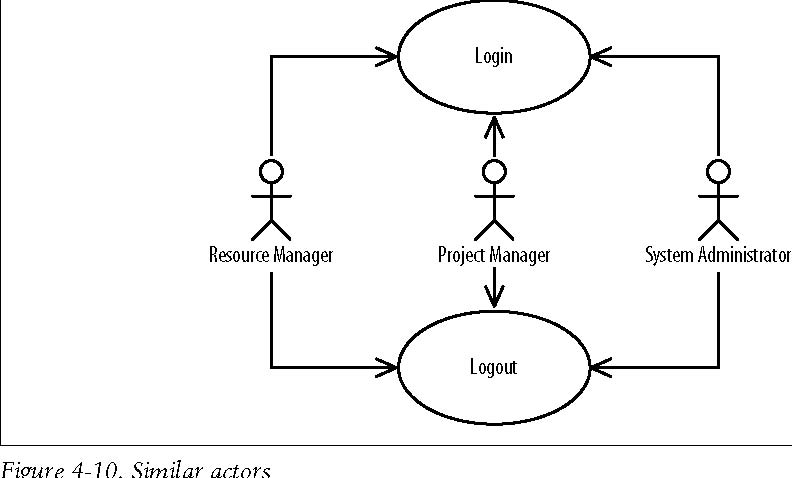








**Generalizations**



**Use-Case Generalizations**

