1
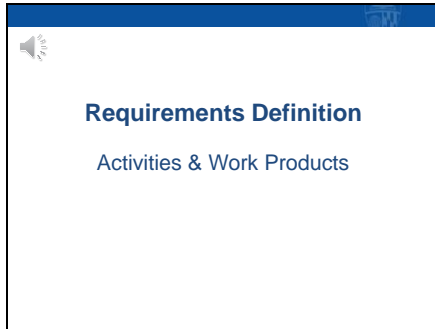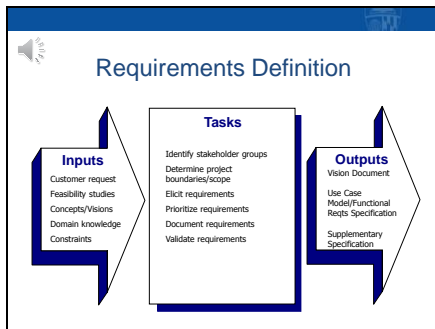
Requirements Definition

Activities & Work Products

In this lecture we'll discuss the key activities in a requirements definition process and also discuss the primary requirements work products.

2

Requirements Definition

Inputs	Tasks	Outputs
Customer request	Identify stakeholder groups	Vision Document
Feasibility studies	Determine project boundaries/scope	Use Case Model/Functional Reqts Specification
Concepts/Visions	Elicit requirements	
Domain knowledge	Prioritize requirements	Supplementary Specification
Constraints	Document requirements	
	Validate requirements	

This input/output model shows the basic inputs, tasks, and work products associated with a requirements definition process.

It is in this process that analysts will identify key stakeholder groups, evaluate the system boundary, and elicit stakeholder needs to identify the project's functional and non-functional requirements. Requirements may also be prioritized and validated as part of this work, and may be packaged into delivery cycles if an iterative or incremental project life cycle model is being followed.

Major deliverable work products produced in this activity typically include a Vision Document, a use case model…if the project is using a use-case driven approach…or a functional requirements specification document if use cases are not being used, and a supplementary specification document.

Depending on the organization, there may be additional work products produced, and the information that is contained in what we are calling a Vision Document and a Supplementary Spec may be packaged into documents with different names…but I'm going to use these two documents because they are a very effective way to capture and organize essential information.

It's worth noting here that if our project is using an object-oriented approach this model would represent the first step in object-oriented requirements analysis…and the work products listed here would represent the user view of requirements. These documents would make no mention of objects, but would serve as the input to the object-oriented requirements analysis step.
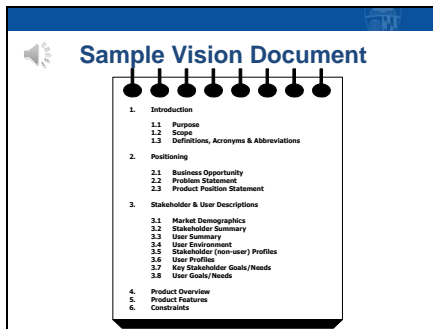
Let's take a look at the kinds of information that go into the Vision Document. A sample Vision Document template is illustrated here.

The Vision Document specifies the stakeholders' view of the product to be developed. It is written in terms of the stakeholders' key goals/needs and desired features. And, it provides the basis for eliciting and specifying the more detailed product requirements.

Some of the document's components, like business opportunity and problem statement, are pretty obvious in their meanings. Some of the components, like market demographics, may not be applicable to every organization and project. This sample template was one that I helped to develop for a software vendor client…so market demographics was very relevant for them.

Section 3 in this document is extremely important. It contains a description of product stakeholders and their needs. Note that there are two types of stakeholders: end user stakeholders and non-user stakeholders. The user stakeholder components were included because our client realized that identification of the actual end-users of the software product is essential to developing a successful product.

A relatively common industry problem that I see much more than I'd like to is failure to actually collaborate with end users during the requirements phase…and it almost

always results in products that are lacking needed functionality.

Many Types Of Stakeholders
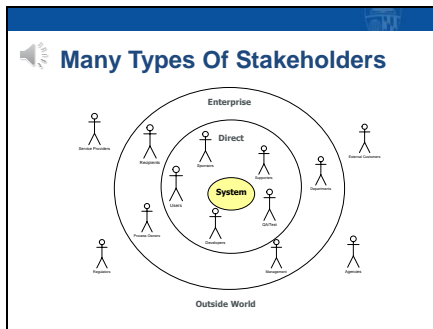
Earlier in the course I talked about project stakeholders and how they contribute to more correct and complete requirements. In an earlier lecture I also defined a stakeholder as any entity that was directly or indirectly impacted by the project at hand.

Using this definition of the term stakeholder is extremely useful in helping a project team identify a more complete set of candidate stakeholders. There can be many types or categories of stakeholder for a project…and this diagram illustrates a sampling of that. I like to call this my layered or concentric ring model…and it can help in identifying a more complete set of stakeholders.

At the center is the system we're gathering requirements for. My first layer of stakeholders is what I call direct stakeholders. These stakeholders are directly impacted by the product we are building, and they consist of, for example, end users, project sponsors, developers, testers, those that will have to support the system when it's in production, and so forth. You can add more categories depending upon the specific organization and product.

The next layer is the enterprise layer. This layer consists of stakeholders that may be indirectly impacted by the system. And the third layer consists of entities outside the organization. This might include service providers

and regulators, as examples.

Some of my clients have used this model as a starting point…and then have added their own specifics. The important thing is to ensure that we identify the most complete stakeholder set possible for a project.
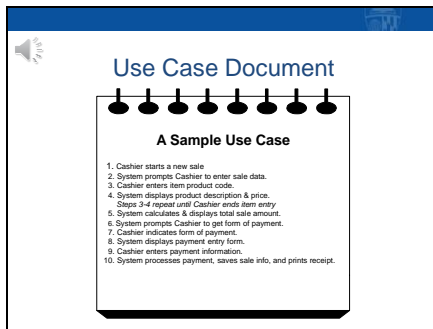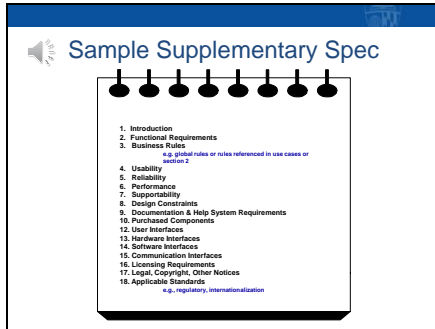
Many projects today are use-case-driven. By that I mean that use cases are used to document the project's functional requirements.

A use case is a different way of documenting functional requirements. It documents those requirements by essentially telling a story that contains related functions the system must perform, the order in which they are performed, and the user interactions that are involved.

This is a partial use case that describes some of the steps and interactions required to process a sale transaction in a point-of-sale system. If you read through the ten steps note that it kind of tells a story about what functions the system must perform when processing a sale. We'll talk about use cases in more detail in a later lecture.

For now, it will suffice to know that one of the key work products produced in the requirements elicitation phase is a set of use cases that document the system's functional requirements.

Sample Supplementary Spec

1. Introduction
2. Functional Requirements
3. Business Rules
   e.g. global rules or rules referenced in use cases or section 2
4. Usability
5. Reliability
6. Performance
7. Supportability
8. Design Constraints
9. Documentation & Help System Requirements
10. Purchased Components
12. User Interfaces
13. Hardware Interfaces
14. Software Interfaces
15. Communication Interfaces
16. Licensing Requirements
17. Legal, Copyright, Other Notices
18. Applicable Standards
   e.g., regulatory, internationalization

Another important document that is produced during the requirements elicitation activity is a Supplementary Specification document. Here's a template for one that I helped develop for a software vendor client.

Section one of this supplementary spec is essentially the same as the first section of the Vision Document. In cases where it is the same, common practice is just to point to that section in the Vision Document instead of regurgitating all the information.

Section two of the Supplementary Spec contains a section for documenting functional requirements that are documented in the traditional way. By "traditional way" I mean by not using use cases.

Section three of the spec contains a section for documenting business rules. The business rules could be global business rules, business rules referenced in use cases, or business rules referenced in section two.

The remaining sections of the spec are used to document a project's non-functional requirements. Not all of these sections would be applicable for all projects, but they represent a pretty good list of possible non-functional requirements.