

What are Requirements

I want to challenge your thinking about requirements. One problem we have in communicating among the various stakeholders in the project team is that we use the same words with different meanings. These meanings vary due to the point of view of the person using the words. One such word is requirement.

We can look at a definition of requirement from some reliable source such as IEEE or IIBA. But that does not really solve the problem. It is in the practical application of the definition that we find our understanding varies from one stakeholder to another.

I want to put the thought to you that all different interpretations of the meaning of requirement are correct, in their appropriate context. We really have to talk about context when we talk about requirements. These are all requirements in some context:

- Use Case Purchase Products Online
- The system shall provide the ability for the user to enter data without using a mouse
- Valid dates are of the form dd/mm/yyyy
-

<code><<interface>></code> Accounts
<code>createAccount () : accountNumber</code> <code>addFundsToAccount (accountNumber, amount)</code> <code>removeFundsFromAccount (accountNumber, amount)</code> <code>removeAccount (accountNumber)</code> <code>balanceAccount (accountNumber)</code>

Rather than telling someone they are wrong about requirements, what we need to do is show the context where each kind of requirement is appropriate. Further, we have to show how the different kinds of requirements are related, and do that in such a way that the various project stakeholders can talk meaningfully about their own requirements and about how their requirements relate to those of other project stakeholders.

What happens if you use the wrong kind of requirement in a particular context? The most obvious problem is that the stakeholder in that context does not have the right information to create requirements of that type. So for example, I should not ask a business person to discuss the details of the operations for a component. Nor should I ask a developer on the team to describe the business processes of the company. Not only that, but I should not ask the developer to write code from the business requirements, nor ask a business person to learn how to use a system by looking at descriptions of functions.

We make a fundamental mistake on a software project when we ask all stakeholders to work from the same form of the requirements. As long as I can show a relationship from one form of requirement to another, there is no problem in having different stakeholders work from different forms of the project requirements.

I will illustrate this concept with a set of examples showing the different kinds of stakeholders and what they need in terms of requirements.

Business Stakeholders

When thinking of business stakeholders, I am thinking of the end user. I want to know what a person needs to do his or her job, and how he or she performs that job. Often we start to capture requirements from the business stakeholders as use cases. Here is a use case that describes a basic process a user might follow:

Apply for a Grant

Precondition: The grant application period is open.

1. The applicant opens the grant application at the Bureau of Grant Applications website.
2. The applicant enters required information into the grant application.
3. The applicant submits the grant application to the Bureau of Grant Applications.
4. The system enters the required information into the database in the state of complete.
5. The system sends a confirmation message to the applicant.

Postcondition: A complete application is stored in the system.

Alternatives:

Grant information is incomplete

In step 3 of the use case, if the grant application does not contain all required information, the system posts a message to the applicant requesting the missing information. The use case continues at step 2.

Save

At any time before the applicant submits the grant application, the applicant may choose to save the application in an unfinished state. The system enters the required information into the database in the state of incomplete. The system sends a confirmation message to the applicant.

Postcondition: An incomplete application is stored in the system.

Cancel

At any time before the applicant submits the grant application, the applicant may choose to cancel the application. The system sends a confirmation message to the applicant.

Postcondition: Nothing is stored in the system.

I can take this use case to my business stakeholders and users and ask them if this use case describes the correct and complete business process. I want the business

stakeholders to focus on what a person is doing. I want to stay independent of technologies as much as possible so that our focus is on the business process. Now I do mention a website in the above use case, which some people will argue is wrong. But if my project is to build a web interface to allow grants to be submitted, then not talking about a website would probably be more confusing than mentioning it. What I really want to have happen is not arguments about including a website in the use case or not, but rather for the business stakeholders to find errors and omissions. Hopefully in that review process someone will notice that while we can save an unfinished grant application, there is no way provided to retrieve and edit it!! Nor is there a way to purge incomplete applications from the system. I also want more information about what that required information is, and if there is any optional information that can be submitted. That is the kind of feedback you really want on this use case.

At this point, since I will have a person using the system, I'd really like to have someone sketch out a user interface or application form. It will be easier to find out and show the required and optional information if it is presented to the users in the form of an interface. This does not have to be fancy! Some sketches on a white board will go a long way, and I can take a digital picture of them to capture the information. This shows the first sketch and some comments from the business stakeholders on the right:

The sketch shows a form with the following fields:

- Name _____
- Address _____
- City, state zip _____
- SSN _____
- Grant applied for Sur [dropdown menu]
- Date of submission _____
- Proposal name _____
- Abstract (600 chars) [text area]

Annotations on the right:

- Red arrow pointing to the dropdown menu: Pull down list of currently available grants
- Red arrow pointing to the Date of submission field: Auto-populated when completed application is saved. Need time of submission as well.
- Text below: All of this information is required or we will not process the application.

I also want to find out any rules associated with the grant process or with the information collected. Some example rules might be:

- A social security number (SSN) must be 9 digits.
- The abstract can be no more than 600 characters. This is an industry standard for the length of an abstract.
- We only need to format addresses for the US because grants can only be submitted by people living in the US.
- We will only process complete grant applications. Complete means all required information has been included.

I can continue this process of eliciting information from the business stakeholders until I feel comfortable that I have captured everything the business stakeholders know about the process of submitting a grant application, the information that is required, and any rules that must be followed.

Will the development team start to work on writing code now? That is very unlikely. While my business stakeholders will likely agree that the requirements are now complete, the development team will say they need a lot more information before they can write code. But the business stakeholders think they have already given me complete requirements! So who is right?

Both the business and technical stakeholders are correct. From the point of view of the business stakeholders, the requirements are complete. From the point of view of the technical stakeholders, the requirements are not complete. So where do the additional requirements come from? The business requirements came from the business stakeholders, so the technical requirements must come from the technical stakeholders. And we will have another group of stakeholders representing standards or regulatory requirements.

Standards and Regulations Stakeholders

I can assume some of the kinds of requirement I will need because of the kind of application we are creating. For example, because this is a web based application, I can expect requirements for privacy and security. There are probably also company standards around the user interface itself, both what it looks like and how it works. Since this is a government agency and money is involved, we also have requirements for auditing and reporting. We will likely have other requirements to use existing systems or to replicate existing functionality.

My business stakeholders may know about these kinds of requirements or may not. The stakeholders I really need to visit now are in Enterprise level organizations. I will look for groups such as Enterprise Architecture, Governance, Corporate Security, Enterprise Information, Finance, Auditing, and Legal.

Some of the requirements I found in my investigation with these groups are these:

- Social Security Numbers must be encrypted.
- There must be a prominent link on every web page to the company's online privacy policy.
- The website must be available 24/7 during the defined open period for grant applications.
- There must be a prominent statement of our lack of liability if the website or server is not available when the person tries to submit his or her application. We are not responsible for a system failure during the submission process.
- We have to have reports of grant submissions categorized by grant and by submitter demographics.

- We have to be able to link an award back to the original grant submission for reporting purposes.
- There has to be optional information collected that tells the person's age, sex, ethnicity, and veteran status.
- In a prominent place on the application web site, we have to include information about who owns the information on the application and who will own the results of the project that the grant is paying for.
- The website must use the published corporate standards for color, layout, and use of standard features such as corporate logo and presentation of error message.
- The project must use existing systems in the implementation of the software, unless those systems do not support some required functionality.
- The project must use existing corporate data stores, unless those data stores do not support some required functionality.
- All submissions must be kept for a period of 7 years after the closing of the grant application period, including incomplete submissions.

This is a good starting point for more investigation. I still need many more details before the development team can do their work. For example, what exactly are the reports? What information do they contain? Do they have a specific layout? Who uses them? How are they consumed – electronically or printed out? Are they displayed to a user or stored in a database? Do they have to be saved? If so, for how long?

For reports, I really like to get a sample layout that shows the required information. If it is a new report, then I will create a sample layout either with a whiteboard sketch (as we did for the user interface) or using a spreadsheet tool. If it is an existing report, I will get a copy of a current report. If some of the information is calculated, then I will need to add notes to the sample layout. These notes will include the formulas to be used for the different fields on the report. If I use a spreadsheet tool, I can mock up the report with the formulas included in the cells that represent the calculated data fields.

Total number of Grant Applications from June 1, 2008 - September 5, 2008		sum of grant totals
Total Applications by Grant		
Grant 1 Name	sum of state totals	
	Alaska	NNNNNN
	Arkansas	NNNNNN
	California	NNNNNN
	Colorado	NNNNNN
	Connecticut	NNNNNN
	Delaware	NNNNNN
Grant 2 Name	sum of state totals	
	Alaska	NNNNNN
	Arkansas	NNNNNN
	California	NNNNNN
	Colorado	NNNNNN
	Connecticut	NNNNNN

Grant 3 Name	Delaware	NNNNNN
	sum of state totals	
	Alaska	NNNNNN
	Arkansas	NNNNNN
	California	NNNNNN
	Colorado	NNNNNN
	Connecticut	NNNNNN
	Delaware	NNNNNN
Total Applications by State		
	Alaska	NNNNNN
	Arkansas	NNNNNN
	California	NNNNNN
	Colorado	NNNNNN
	Connecticut	NNNNNN
	Delaware	NNNNNN
Total Applications by Zip Code		
	01234	NNNNNN
	15217	NNNNNN
	95051	NNNNNN
Total Applications by Sex		
	Male	NNNNNN
	Female	NNNNNN

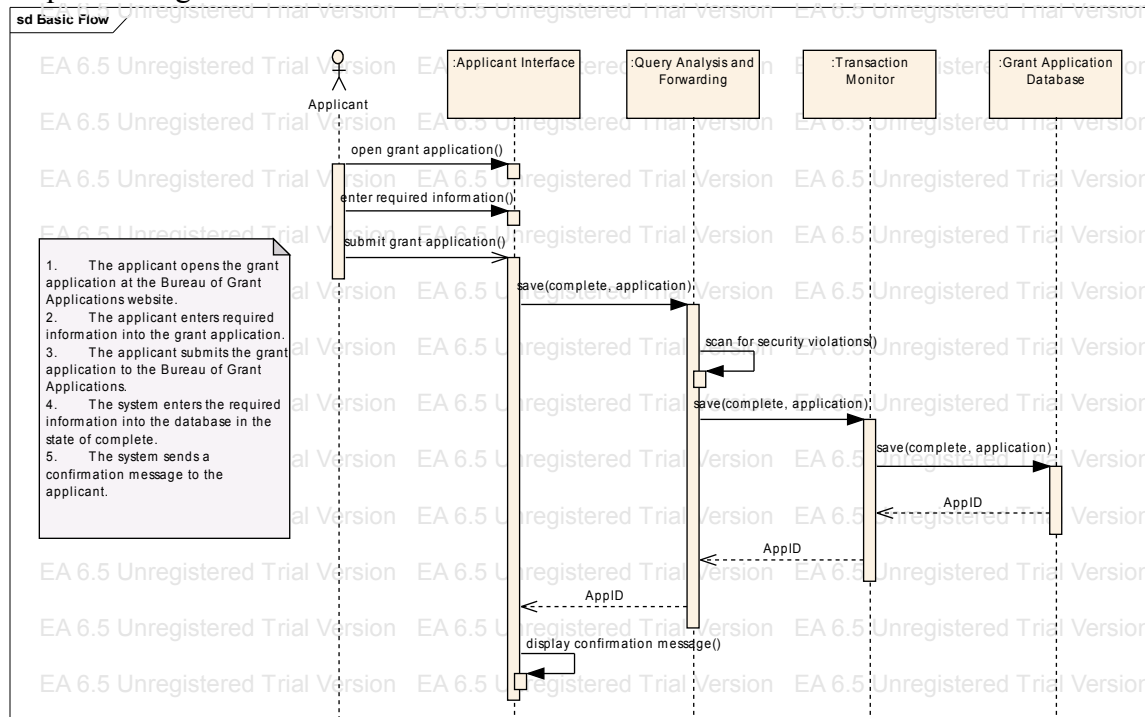
Are we there yet? Are we ready to write code? It is unlikely that we are ready to write code. There is still information that the developers will require before they can code and test the system.

Technical Stakeholders

In order to create code, there is technical information that has to be discovered. This information includes things such as which components from the component architecture will be used on this project, what languages and technologies will be used in the project, the data model for the information that has to be stored, and any patterns or coding standards that must be followed.

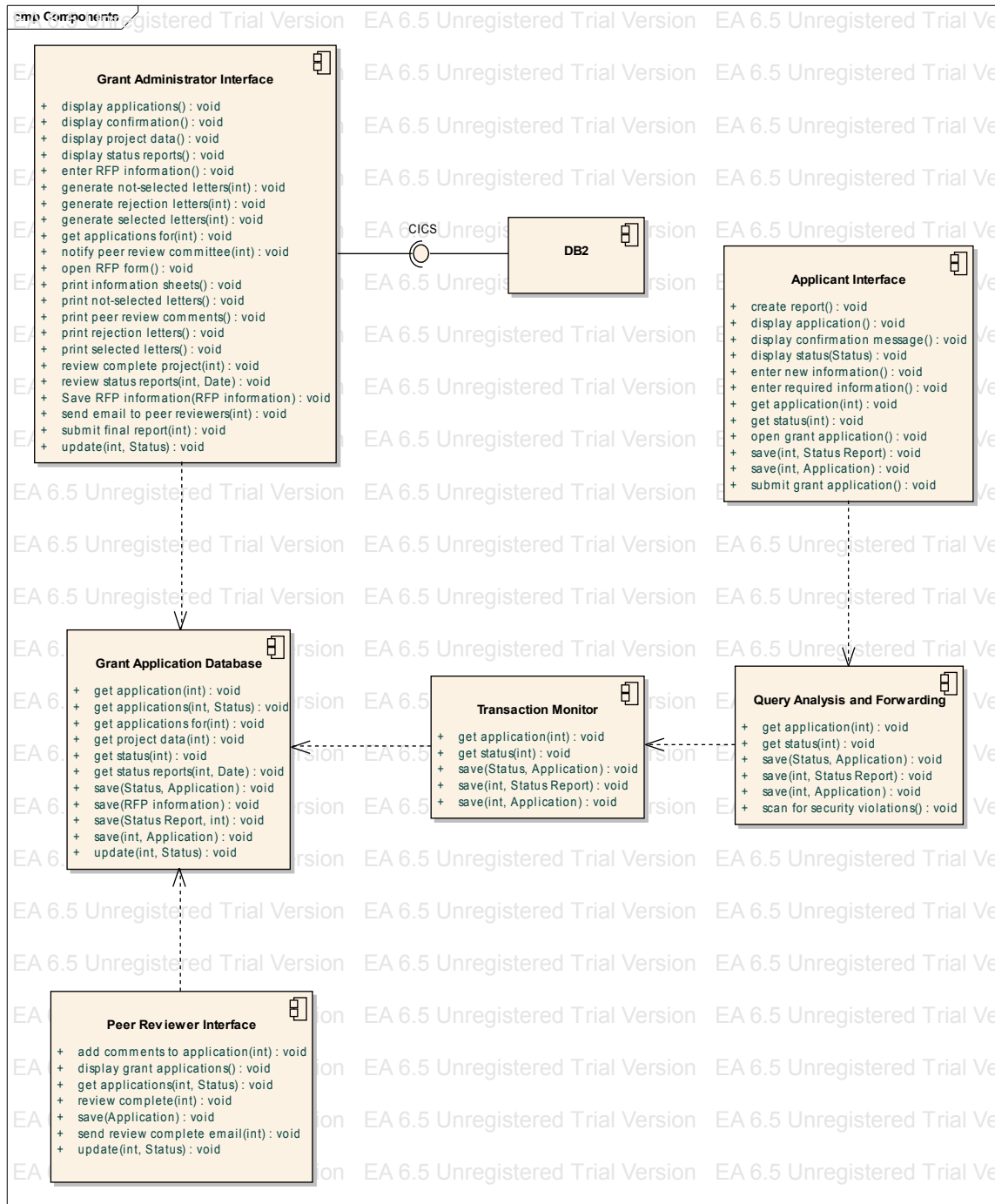
I have to be familiar with the requirements collected from the business stakeholders and the standards and regulations stakeholders, and also be familiar with the Enterprise architecture to determine what components will be used in creating the new software. Some components will represent existing software or systems that can be used as-is, some components will represent existing software or systems that will need some modification, and some components will be new. The new components may be something that the developers create, or may represent software or a system that we will purchase. Along with identifying the components, I have to identify the company coding standards and patterns that will be used on the project.

A great way to do this is to create diagrams showing exactly which components will implement which requirements. One kind of diagram I can create to show this is a UML Sequence diagram.



This sequence diagram also is the place to put required patterns. In this diagram I see the use of a pattern to store data: The application Interface sends the complete application to a component which looks at formatting and checks for security violations (such as embedded SQL in the text fields). Then the complete application goes to a transaction monitor, and then it goes to the database. I would expect to see the same pattern of messages every time a user of the application interface wants to store data.

I can create a UML Component diagram which focuses more on the individual components (rather than focusing on the flow as the sequence diagram shows). This also gives me a place to show requirements that are not part of a use case. So you see that the Query Analysis and Forwarding component has a function to scan for security violations. That was not part of the business stakeholder's use case, but came from corporate security requirements.



In doing the exercise of mapping the requirements to components, I want to be thorough and make sure every requirement is being implemented somewhere. I can also indicate technologies in this diagram, as you see from the CICS interface in the top center which connects to a DB2 database.

At this point I need to look at each component and identify the technologies that will be used to implement it. This may include languages such as C++, Java, or Python, and whole technologies such as Pega Rules Engine. It will also include communication paths

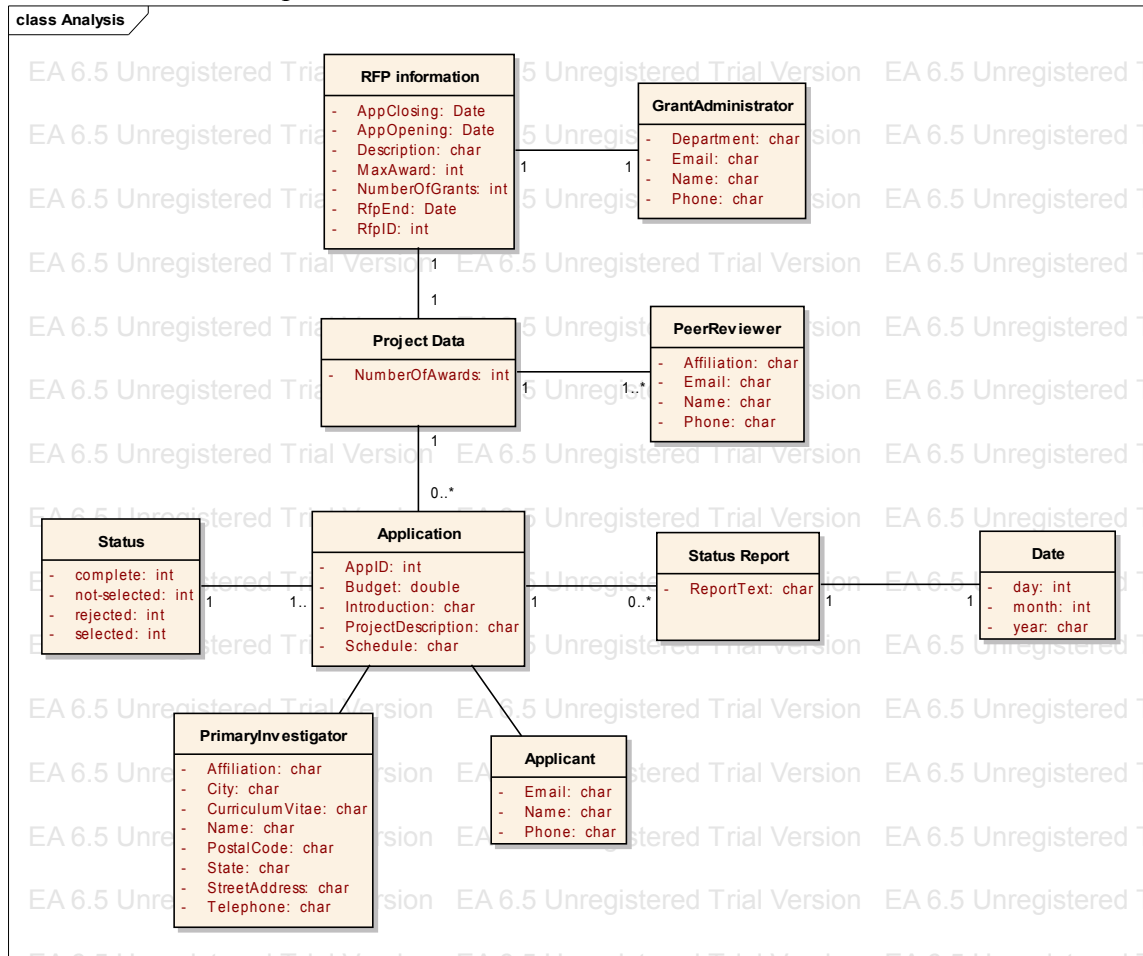
for transferring information between components on different systems. In addition, I need to be explicit about which components need to be implemented and which will be used as-is. I may create a separate document for each component that gives this information.

For example, for the Grant Application Database, I need to determine and document the at least the following:

- The corporate standards say I have to use SQL Server for this database. I need to see if any other software is required to handle queries and return the requested result. Based on the users of the database, I need my database to interact with applications and users on a mainframe, Unix, and Windows. Some people will use Crystal Reports to do canned and ad hoc queries against the data. There is an existing database that is being retired. The data from this database needs to be moved to the new SQL Server database. Any reports that use the old database have to be changed to use the new database.
- I have to determine what hardware we will run the server on. We will have an unknown number of users log into the web site to upload grant applications. This number will spike to maximum load in the hour before the grant application time frame ends. Experience suggests that 5000 people will apply for grants in any given cycle, and that 2500 of them will try to upload applications in the last hour. In addition, grant administrators, reviewers, and auditors will need to access the data. We want to deny access to those users in the last day of the grant application time frame. When the application period closes, we have to lock grant applicants out of the application, and allow the grant administrators, auditors, and reviewers access to the data. At the most, we will have 100 concurrent users of the system at this time. The response time for these users should be 10 seconds or less to respond to a query for a grant application. A grant application is always less than 5 mb of data. Updates of the data are just status fields, so the amount of data per user is negligible. Auditors may run reports that show consolidated numbers for 1/7 of the records in the database. They may also wish to review individual grants, and may request up to 20 grant applications at a time. Data has to be retained for 7 years. The database server must be up 90% of the time. Data will be backed up at least nightly to another server.
- I need a data model for the database. I can get information on what data is required by looking at the information collected on the sample application form, and by looking at the use case for other information I need to save with the application. I need to look at the sample reports to see if all the information I have discovered will support the creation of the reports.
- I have to look at the standards and regulatory requirements to discover if there are policies for use of the data. Some examples include whether to allow caching or not, web services or not, agents or not, stored procedures or not, and any access rules for agency personnel (who is allowed to access the data, update the data, and maintain the database). There may also be data access patterns I am required to use.

I mention a data model in the example of the component. A data model is typically created using Entity-Relationship Diagrams (ERD), though it can also be created using

UML class diagrams. These diagrams show the tables and records of a database, and their relationships. When creating the tables and relationships, I have to consider the information that is required, as well as how the data will be accessed over time.



Along with systems, technologies, and data models, the user interfaces have to be designed. We made sketches of the user interface and report layouts earlier, but now I need to take that information and the corporate standards for web applications, and create the actual design of the user interface.

This activity may be done with wireframes or storyboards. This will be a more realistic looking user interface than our sketches. Its purpose is to show exactly what the user interface looks like and how it behaves.

Now the developers have the requirements for their work. They still have to determine exactly how to make all this happen. As a developer, I will look at my required technologies and systems to find out what already exists that I can use. Assuming I am using Java, I will determine what Java classes I can use, and what new classes and functions I have to create. I will also write the code to communicate with other components of this project. I will test all my code, and give it to a build engineer to put my code together with my team mates code to test it all together.

Who Does the Requirements Work

When working with the business stakeholders, you may have a Business Analyst do the work. A Business Analyst knows how to communicate with business people in their language. This person will very likely also collect the initial list of requirements from the various Enterprise organizations, but not necessarily the details of the requirements. This person needs to have excellent people and organizational skills.

When working with the standards and regulations stakeholders, you may have a System Analyst do the work. A System Analyst is someone who knows the systems and software and can research what is required from that point of view. So for example, the System Analyst would know where to go to pull sample report layouts and to find the formulas for creating them. This person needs to know the systems and where to find information in the systems. This person needs access to the systems that contain the information.

There are a number of people who will determine and document the technical requirements. The project Architect is responsible for determining the components, patterns, and standards to be used on the project. One or more project Designers will determine what technologies and computer languages will be used on the project. A Database Designer will determine the requirements around information and the persistence of data. Finally, a User Interface Designer will determine all the details around the user interface, what it looks like, and how it works.

Geri Winters
Durham, NC
March 14, 2009

<http://www.writingusecases.com>