Why Requirements?

Why Requirements?

Process & Requirements

Requirements combined with processes are the backbone of any successful software project.

Why Requirements?

Process & Requirements

Requirements combined with processes are the backbone of any successful software project.

- A robust software development process is your launchpad for building great software.

Why Requirements?

Process & Requirements

Requirements combined with processes are the backbone of any successful software project.

- A robust software development process is your launchpad for building great software. What are the initial software requirements?
  - The first step in all development processes is to understand what you are creating.
  - Requirements are a set of specific descriptions of your client's needs.

Why Requirements?

Example of requirements for a great social network app

- The user could store user profiles and their data,
- send messages to other users,
- view other users' profiles,
- create status updates, and
- view other users' status updates.

Requirements generate the features of a software product.

-> Ambiguity requirements -> Unclear features to develop

Why Requirements are important?

What happens when you lose focus on those requirements?

Example of unclear and ambiguity requirements

How to translate the following request into a requirement?

- A simple request: "users can talk to each other through the app"

Why Requirements are important?

What happens when you lose focus on those requirements?

Example of unclear and ambiguity requirements

How to translate the following request into a requirement?

- A simple request: "users can talk to each other through the app"

confusion for developers: are users supposed to be able to send

- voice notes [text messages / brain waves?] to one another?

Why Requirements are important?

What happens when you lose focus on those requirements?

Example of unclear and ambiguity requirements

How to translate the following request into a requirement?

- A simple request: "users can talk to each other through the app"

confusion for developers: are users supposed to be able to send

- voice notes [text messages / brain waves?] to one another?

And then?

Why Requirements are important?

What happens when you lose focus on those requirements?

Example of unclear and ambiguity requirements

How to translate the following request into a requirement?

- A simple request: "users can talk to each other through the app"

confusion for developers: are users supposed to be able to send

- voice notes [text messages / brain waves?] to one another?

And then?

- need to get more details avoiding confusion by properly eliciting and expressing requirements
- refine requirements to detect potential errors

-> By clarifying ideas, development becomes focused and efficient.

Example: A social network app requirement

One of your developers has begun developing

- A social network app with a requirement:
- "Users must be able to send messages as a guest."

Requirement interpretation

The developer might take this requirement and design a system in which users have the ability to either log in

- to their own account
- or use a guest account.
- What about a single guest account?
- Consequences when many different users have access to one guest account?

Quiz

Quiz on "Why requirements are important in software development?"

- You're working with your client to elicit a set of requirements for a software project.
- The client produces a napkin on which he has
  - sketched out an app
  - with a few sparsely worded features.
- He says, this is the product we want.

If you accept this document as your final set of requirements, which of the following are potential risks to your project?

- A. The user interface problems may not be noticed until software is built.
- B. The end product may not meet the client's expectations.
- C. The client has too much input over the look and feel of the project.
- D. Technical limitations will be detected late in the project and be expensive to work around.

Quiz

If you accept this document as your final set of requirements, which of the following are potential risks to your project?

- ✓ A. The user interface problems may not be noticed until software is built.
- ✓ B. The end product may not meet the client's expectations.
- ✗ C. The client has too much input over the look and feel of the project.
- ✓ D. Technical limitations will be detected late in the project and be expensive to work around.

Quiz

If you accept this document as your final set of requirements, which of the following are potential risks to your project?

- ✓ A. The user interface problems may not be noticed until software is built.
- ✓ B. The end product may not meet the client's expectations.
- ✗ C. The client has too much input over the look and feel of the project.
- ✓ D. Technical limitations will be detected late in the project and be expensive to work around.

Problems of unrefined requirements

- lead to detecting issues late in the project.
- – fixing those issues can be costly
- in the risk of not understanding client's expectations.