Requirements

Definitions and Issues

Refs: John Vu (CMU)

What is software engineering ?

Practical Software Engineering

- Software engineering is training focusing on creating cost-effective solutions to real world problems by applying engineering knowledge to build quality software systems
- Software engineers learn how to make decisions in designing and implementing solutions under constraints of limited time, knowledge, and resources.
- The most important practice in software engineering that provides the greatest benefit is requirements engineering

How many phase in software process ?

What are requirement ?

What Are Requirements?

- □ A condition or capability needed by a user or customer to solve a problem or achieve an objective.
- □ A condition or capability that must be met by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.
- □ A document representation of a condition or capability

Requirements Are …

- Requirements are descriptions of the necessary and sufficient properties of a product or service that will satisfy the customer’s need.
- Software requirements are descriptions of the necessary and sufficient properties of the software that must be met to ensure the product achieves what it was designed to accomplish for its customers or users

Software Requirement - 1

- Every software project has users who rely on the software to do something for them.
- The time spent understanding and writing down what users need is very important.
- If software people do not have well written requirements that users agree to, how can they develop software that satisfy those users?
- If you do NOT write down requirements but assume you know requirements, you may develop something users do not want

Software Requirement - 2

- The “ TO DO” List of the Project Team.
- The List of “WHAT” Customer’s need.
- The List of “WHAT” the software must do to satisfy its Customers.
- The List of “WHAT” components must be built.
- The List of “WHAT”each component must “DO” and “HOW” they will “INTERACT”.

Software Requirement - 3

- Requirements describe the behavior of the software as seen from the customer’s perspective.
- Requirements serve as a communications channel between customers, users and project managers who are concerned with the development of software products or services

Requirements Engineering

- A method of obtaining a precise formal specification from the informal and often vague requirements with customers.
- The science and discipline concerned with analyzing and documenting requirements. It comprises needs analysis, requirements analysis, and requirements specifications.

Belief

- Firm requirements ?
- If can get req → build perfect products ?

Wrong Belief

- Many people believe that for every project there is a set of firm requirements.
- If they can get them, they can build them and produce a perfect product or solution.
- Students are often taught that customers will give them requirements just like a professor gives them assignments and all they have to do is to build the software accordingly.

Wrong Belief … again

- Many people believe customers will clearly provide:
  - Functional requirements
  - How they want the work to be done
  - How it will be used
  - Performance & Scalability
  - System boundary (Scope)
  - Operating environment (Domain)
  - Verification criteria

What are customers actually give us ?

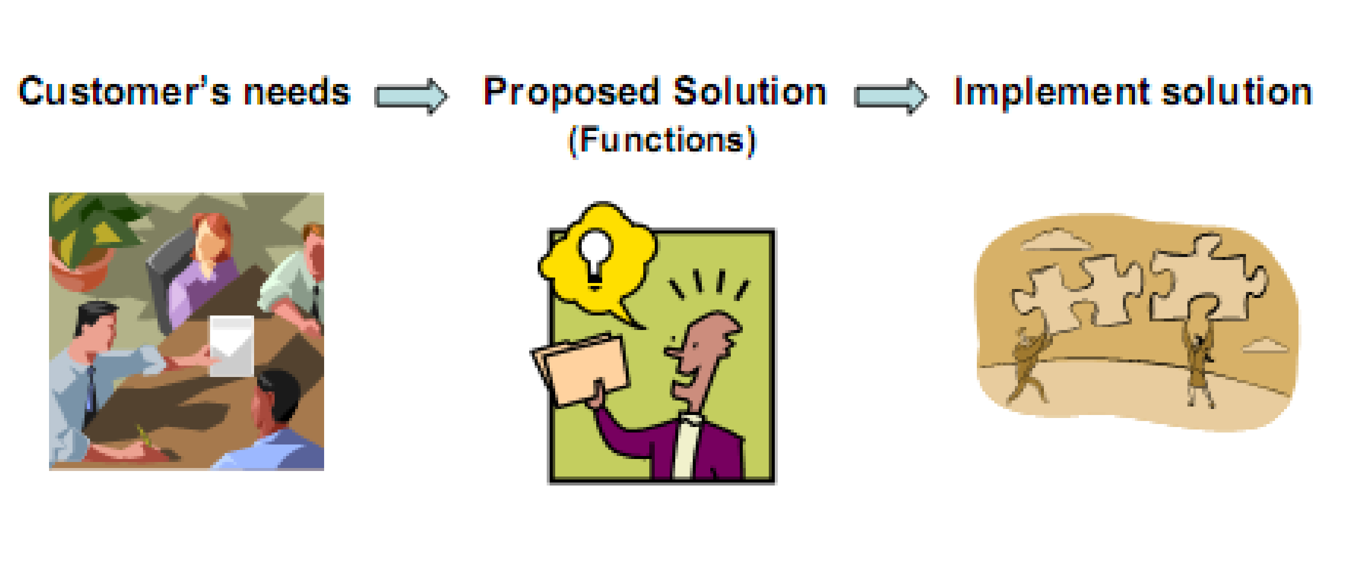
Actually …

- Customers will provide:
  - A wish list of what they would like to have.
  - A solution to their problems without knowledge about how it might be implemented.
  - A vague description that limits implementation.
  - A technology that they read from newspapers.
  - Changes as they often change their minds.
  - Strict budget & schedule

Why Is It So …?

- Many customer expectations are NOT based on needs but wants
- University training is still focusing on solving problems NOT identifying problems.
- Most software engineers do not receive adequate training on requirements engineering.
- Many software engineers want to work on solutions rather than take time to understand the problem (code first, ask questions later).

The Academic View



Customer's needs → Proposed Solution (Functions) → Implement solution

The Academic View

- This simple view only works when there are:
  - Unlimited resources
  - Unlimited time
  - Unchanging requirements
  - Great working environment
  - Perfect communication
  - No constraints

The real world view

- Experienced engineers know that there are:
  - Insufficient resources
  - Insufficient time
  - Ever-Changing requirements
  - Highly political working environment
  - Imperfect communication
  - Financial constraints
  - Schedule constraints
  - Other constraints …

Why Requirements Engineering?

- Failure to develop good requirements is the major cause for software project failures
- Lack of knowledge of customer’s business process contributes to the failure of requirements engineering

Requirements Issues

- Failure to understand customers’ needs or their business problems is the major cause for software project failures.
- Software people must learn to listen to the “voice of customers” and understand their business process during requirements gathering.

Requirements Defects

- Requirements defects are poorly defined requirements, errors in requirement caused by incorrect, incomplete, missing, or conflicting requirements.
- Defective requirements may result in:
  - Project failures
  - Expensive rework
  - Cost overruns
  - Poor quality
  - Late delivery
  - Dissatisfied customers
  - Demoralized developers

Customers Vs Stakeholders

Who is the Customer?

- A customer is an individual or organization who derives either direct or indirect benefit from a product.
- A software customer is an individual or organization who request, pay for, select, specify, use or receive output generated by a software product
- Sometimes the term “customers” is generalized to “stakeholders”. However, not all stakeholders are customers or users but they do have influences over the software development.

Who Are Stakeholders?

- To build a useful software, we need to know its requirements.
- To know its requirements, we need to know the stakeholders' needs
- Stakeholder is a person or group that has an **interest** in the software and can **influence** the software requirements or can be **impacted** by the software product.

Who are Stakeholders?

- Customers
- Users
- Analysts
- Developers
- PMs
- Manufacturing people
- Sales, marketing, field support

Questions

- Do you know who your stakeholders are?
- Who else should be considered a stakeholder?
- How many stakeholders are there?
- How familiar are stakeholders with the business?
- What level of skills and knowledge do they have?
- What is a successful solution worth to these stakeholders?
- How much time do we have to solve this problem?

Issues With Stakeholders

- □ Different perspectives on the software project being developed.
- □ Different backgrounds can cause communication problems.
- □ Different objectives which influence views on the requirements.
- □ Different abilities to express requirements and document them.
- □ Different involvement, some can make decisions and others may not.

Never assume that all stakeholders share a common notion of what requirements are.

Calling all stakeholders …

- The first step in requirements engineeringis to identify everyone who should participate in defining the requirements
- Each person has **different perspectives** on requirements:
  - Customers
  - Users
  - Indirect users/Support personnel
  - Managers
  - System engineer/Sales & Marketing people
  - Software developers

Prioritize Stakeholders

- Not all stakeholders are equally important, so it is essential to prioritize the identifiedstakeholder roles into:
  - Critical
  - Major
  - Minor

Key Requirements Concept

- Software engineers must answer these
- questions:
  - **Who** are stakeholders?
  - **What** do they want?
  - **Where** could it work?
  - **Why** do they want it?
  - **How** will we know?
  - **When** should we build it?

The Facts Are …

- If you do not get the requirements right, it doesn’t matter how well you execute the rest of the project.
- Requirements development is a discovery and invention process.
- Requirements change happens.
- Stakeholders are not always right, but they always have a point.
- Stakeholders involvement is the most important factor to the project.

Key Concepts

- Know who your stakeholders are.
- Understand stakeholders’ needs.
- Transform stakeholders’ needs into business requirements.
- Specify requirements based on priority

Summary

□ Requirements Engineering is the first opportunity to “mess up” the project.

□ Many software developers are not trained in Requirements Engineering.

□ Requirements Engineering activities must start early in the project.

□ Software developers must understand stakeholders’ roles.

□ Requirements are the primary reason for most project failures

Questions

- Knowing your Stakeholders’ Business:
  - What’s their product or service?
  - Who is “ *their*” customer?
  - How do they reach “ *their*” customer?
  - How do they make money (or get rewarded)?
  - *Do they have the right personnel to do their job?*

Questions

- Knowing your business:
  - What’s your product or service?
  - Who is *your* customer?
  - How do you reach *your* customer?
  - How do you achieve success?
  - *Why is your team able to accomplish this?*