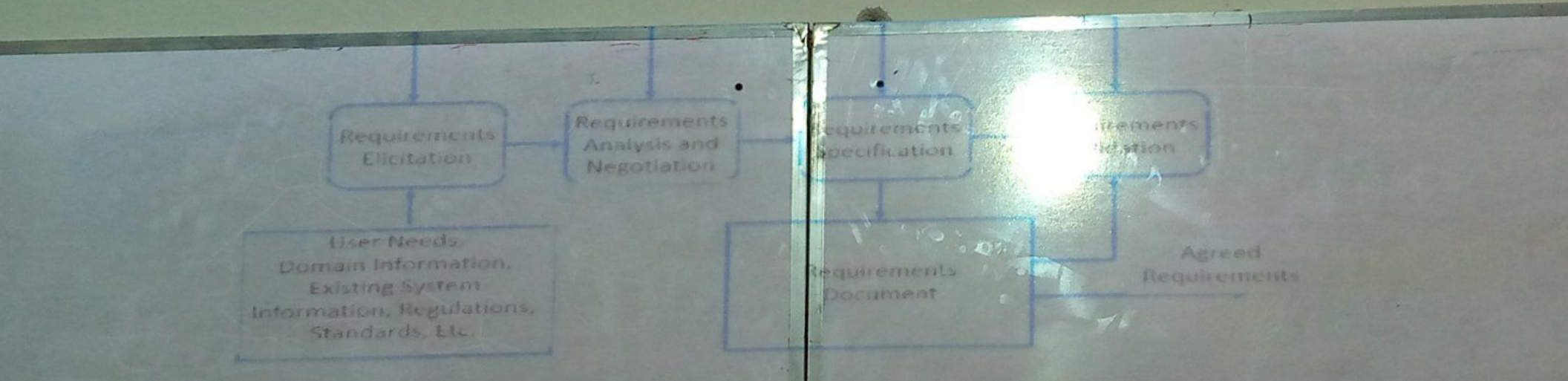
Requirements Engineering Process



Requirements Management

- The process of managing changes to the requirements for a system
- In this lecture, we'll talk about the reasons for changes in requirements and how to manage them

Requirements Management and Traceability

- Requirements cannot be managed effectively without requirements traceability
  - A requirement is traceable if you can discover who suggested the requirement, why the requirement exists, what requirements are related to it and how that requirement relates to other information such as systems designs, implementations and user documentation

Change - A Constant

- There is nothing permanent except change
  - Heraclitus (500 B.C.)
- No matter where you are in the system life cycle, the system will change, and the desire to change it will persist throughout the life cycle
- Software is like a sponge due to its susceptibility to change

Changing Requirements

- All stakeholders want to change requirements, due to different reasons
- Studies have shown that very significant percentage of delivered defects can be traced back to changing user requirements
- A major issue in requirements engineering is the rate at which requirements change once the requirements phase is "officially" ended
- This rate is on average 3% per month in the subsequent design phase, and should go down after that
- This rate should come down to 1% per month during coding
- Ideally, this should come down to no changes in testing, however, this is very rare

Sources of Change

- New business or market conditions dictate changes in product requirements or business rules
- New customer needs demand modification of data produced by information systems, functionality delivered by products, or services delivered by computer-based system
- Reorganization or business growth/downsizing causes changes in project priorities or software engineering team structure
- Budgetary or scheduling constraints cause a redefinition of the system or product

Why All This Modification?

- As time passes, all constituencies know more
  - About what they need
  - How to get it done and still make money
- Statement of the fact: most changes are justifi

Managing Changing Requirements ???

- Following quality assurance mechanisms can limit the damage done by changing requirements
  - Formal change management procedures
  - State-of-the-art configuration control tools
  - Requirements reviews

Main Concerns in Requirements Management

- Managing changes to agreed requirements
- Managing the relationships between requirements
- Managing the dependencies between the requirements document and other documents produced in the systems engineering process

CASE Tools for Requirements Management

- Requirements management involves the collection, storage and maintenance of large amounts of information
- There are now a number of CASE tools available which are specifically designed to support requirements management
- Configuration management tools may be adapted for requirements engineering

Stable and Volatile Requirements

- Requirements changes occur while the requirements are being elicited, analyzed and validated and after the system has gone into
- Some requirements are more stable, while others may be more subject to change than others
- Stable requirements are concerned with the essence of a system and its application domain. They change more slowly than volatile requirements
- Volatile requirements are specific to the instantiation of the system in a particular environment and for a particular customer

Requirements Change Factors

- Requirements errors, conflicts and inconsistencies
- Evolving customer/end-user knowledge of the system
- Technical, schedule or cost problems

Requirements Change Factors - 2

- Changing customer priorities
- Environmental changes
- Organizational changes

Types of Volatile Requirements

- Mutable requirements
- Emergent requirements
- Consequential requirements
- Compatibility requirements

Mutable Requirements

- These are requirements which change because of changes to the environment in which the system is operating

Emergent Requirements

- These are requirements which cannot be completely defined when the system is specified but which emerge as the system is designed

Consequential Requirements

- These are requirements which are based on assumptions about how the system will be used. When the system is put into use, some of

Compatibility Requirements

- These are requirements which depend on other equipment or processes

Summary

- Requirements change is inevitable as customers develop a better understanding of their real needs and as the political, organizational [illegible] system must be instituted, changes.
- There are Stable and volatile requirements
- Types of volatile requirement include mutable requirements, emergent requirements, consequential requirements and compatibility requirements

Requirements Identification

- It is essential for requirements management that every requirement should have a unique identification

  The most common approach is requirement numbering based on chapter/section in the requirements document

- Problems with this are:
  - Numbers cannot be unambiguously assigned until the document is complete
  - Assigning chapter/section numbers is an implicit classification of the requirement. This can mislead readers of the document into thinking that the most important relationships are with the requirements in the same section

Requirements Identification Techniques

- Dynamic renumbering
- Database record identification
- Symbolic identification

Database Record Identification

- When a requirement is identified it is entered in a requirements database and a database record identifier is assigned. This database

Storing Requirements

- Requirements have to be stored in such a way that they can be accessed easily and related to other system requirements

Requirements Storage Techniques

- In one or more word processor files
- In a specially designed requirements database

Word Processor Documents: Advantages

- Requirements are all stored in the same place
- Requirements may be accessed by anyone with the right word processor
- It is easy to produce the final requirements document

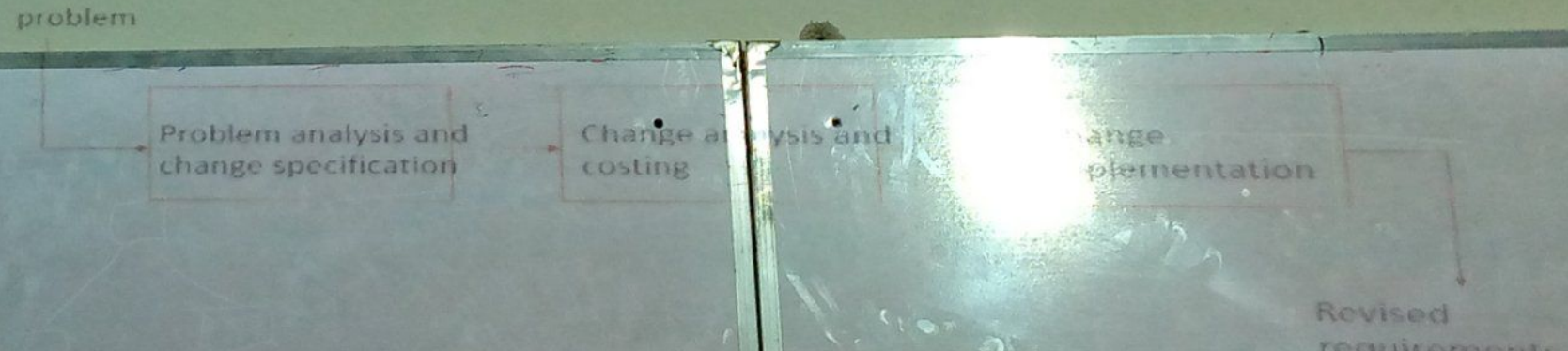
Word Processor Documents: Disadvantages -

- Requirements dependencies must be externally maintained
- Search facilities are limited
- Not possible to link requirements with proposed requirements changes
- Not possible to have version control on individual requirements
- No automated navigation from one requirement to another

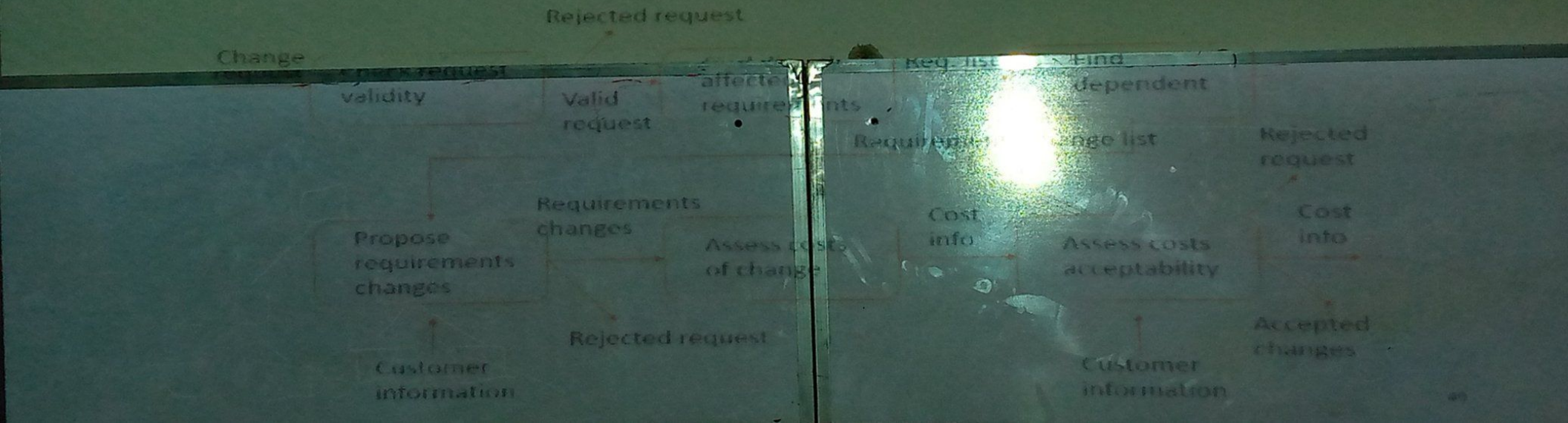
Requirements Database Choice Factors - 1

- The statement of requirements
- The number of requirements
- Teamwork, team distribution and computer support
- CASE tool use
- Existing database usage

Change Management Stages



Change Analysis and Costing Process



Software Requirement Engineering (SE-211)

Lecture 26: Requirements Traceability

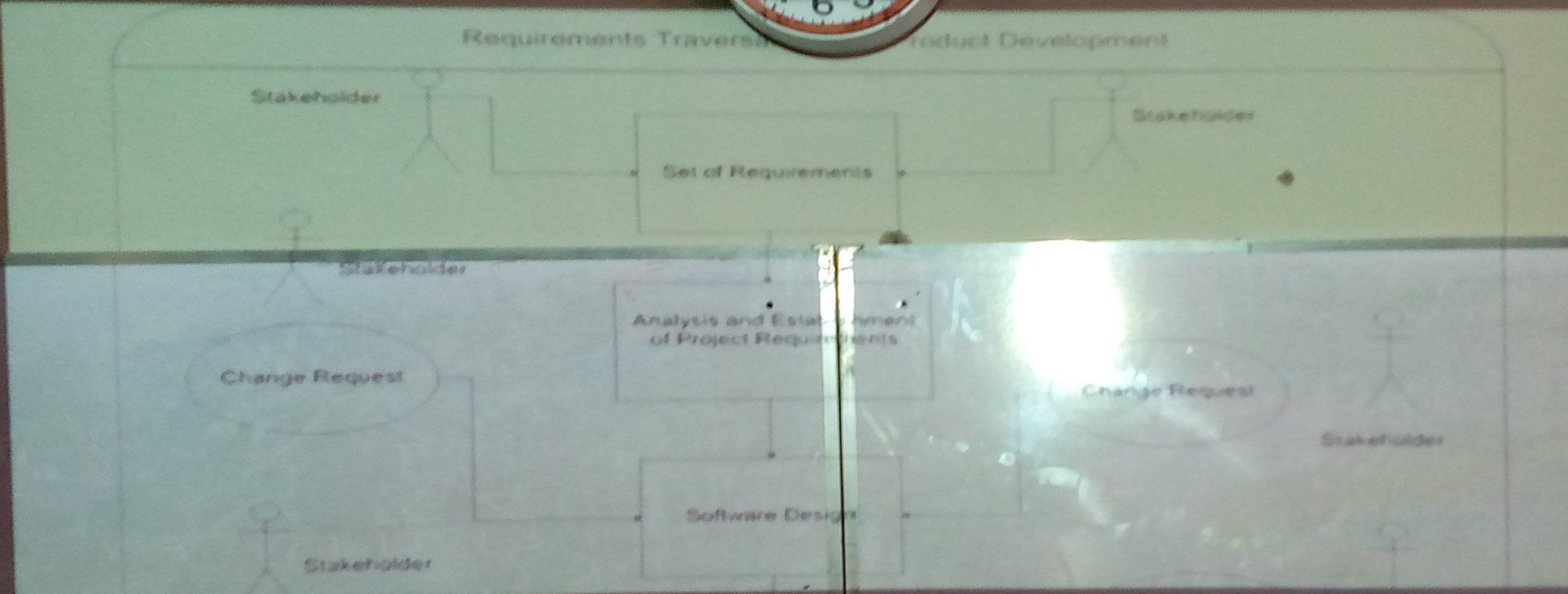
Chapter 6: Requirement Engineering and Management
Chapter 9: Requirement Engineering a Good Practice Guide

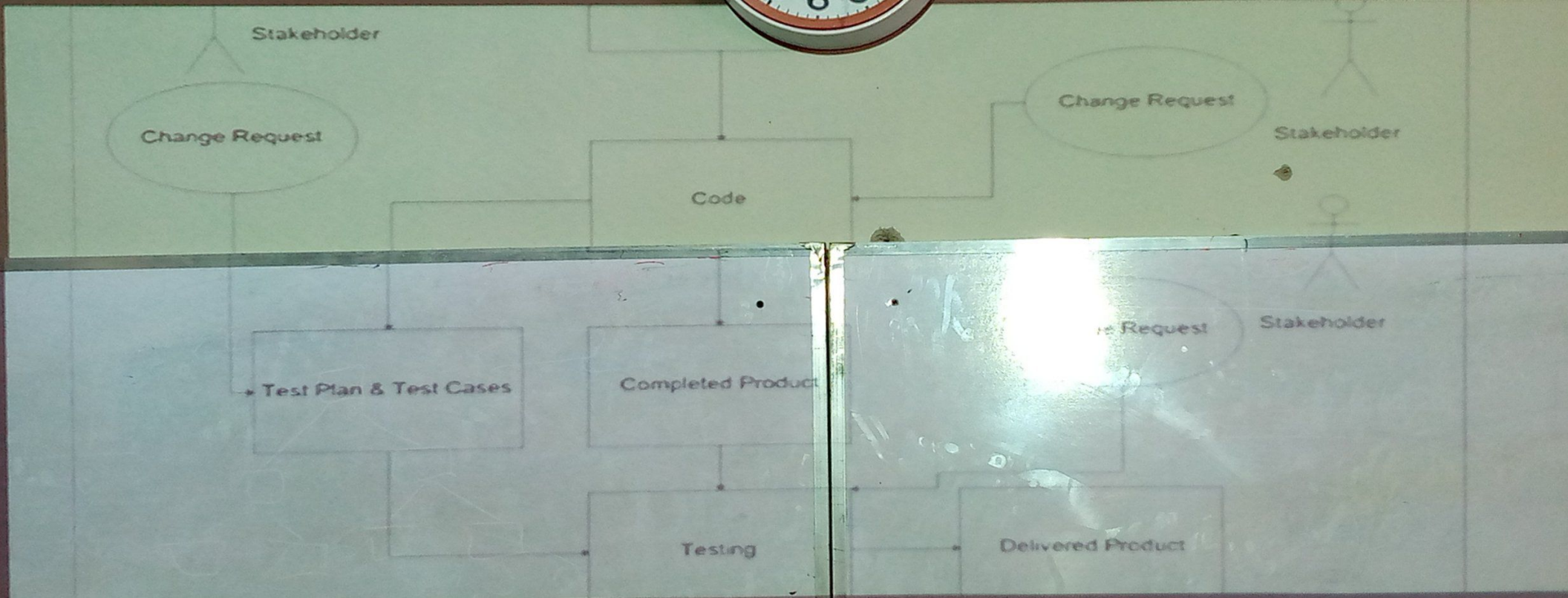
Requirements Traceability

- Requirements tracing involves identifying the requirement in all the software artifacts including information artifacts and code artifacts.
- Refers to ability to describe and follow the life of a requirement, in both a forwards and backwards direction
- That is from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases

Tracing Requirements

- It is important to trace requirements both ways
  - Origin of a requirement
  - How is it implemented
- This is a continuous process


Requirements Traversa
roduct Development
Stakeholder
Set of Requirements
Stakeholder
Stakeholder
Analysis and Estab
of Project Requir
Change Request
Change Request
Stakeholder
Software Design
Stakeholder


Stakeholder
Change Request
Change Request
Stakeholder
Code
Test Plan & Test Cases
Completed Product
Request
Stakeholder
Testing
Delivered Product

Classifications of Requirements Traceability

- Backward-from traceability
- Forward-from traceability
- Backward-to traceability
- Forward-to traceability

A Generic Traceability Table

	A01	A02	A03	Aii
R01				
R02	✓	✓		
R03		✓		✓
Rnn	✓	✓		