



Lesson Objectives

- Understand CRUD
- Life Cycle of Data
- Examples on CRUD using CRUD matrix
- Test situations, test cases, Coverage
- Examples on situations



10.1 Description

Covering of all actions
(Create, Read, Update, Delete)
on all entities

The data that are stored and maintained in the system under test have a *life cycle*. This starts when an entity is created and ends when it is removed. In between, the entity is used by updating it or consulting it.

10.2 Life cycle of data

- Test cases are created based on the life cycle of data
- Lifecycle starting with “C” ending in “D”:

C(reate)	create
R(ead)	use, but don't change
U(pdate)	use and change
D(elete)	delete
- Overview by means of CRUD-matrix:
 - For each function is determined
 - Which entities are used?
 - Which actions (C,R,U,D) are being performed on these?

An overview of the life cycle of the data, or entities, is obtained with the aid of a “*CRUD matrix*”. This is a matrix in which the entities are shown horizontally on the axes and the functions vertically. The matrix is filled in using the letters C(reate), R(ead), U(pdate) and D(elete). If a function executes a particular action in connection with an entity, this is shown in the matrix by means of a C, R, U and/or D.

10.2 Life Cycle Data

Example Situation

- Booking seats for a concert
 - Part of a system deals with the booking of concert seats for clients
 - Available seats and prices depend on:
 - Membership of client
 - Row of the seat

10.2 Life Cycle Data Example

Data items and functions

- Booking seats for a concert
- The following data is of importance:
 - The personal data of the client
 - Membership data of the client
 - The seat reservations of the concert
 - Prices of different rows
- The following functions are relevant:
 - Manage client
 - Manage membership
 - Order ticket
 - Cancel ticket
 - Request information on seats

10.2 Life Cycle Data Example

Creating CRUD-matrix

- Booking seats for a concert The relevant part of the CRUD-matrix is:

functions	Data item	client	subscription	reservation	row
Manage client		CRUD			
Manage membership		RU	CRUD		R
Order ticket			R	C	R
Cancel ticket			R	D	
Request Information on seats		R		R	R

10.3 Test situations, test cases, coverage

- Every processing step (C,R,U,D) is a test situation.
- Test Cases go through the lifecycle (from C to D)
- Less thorough coverage (default)
 - Every C,R,U and D executed at least 1x
- More coverage variations
 - After every U execute all R-actions
 - Similarly after every C, U and D

10.4 Example Test situations

- For data item “client”
 - Manage client (C)
 - Manage client (R)
 - Manage client (U)
 - Manage client (D)
 - Manage membership (R)
 - Manage membership (U)
 - Request information on seats (R)
- etcetera

10.4 Example

Test case, less thorough coverage

- For data item “client”
 - Manage client (C)
 - Manage client (R)
 - Register membership (needed for next actions)
 - Manage membership (R)
 - Manage client (U)
 - Manage membership (U)
 - Request information on seats (R)
 - Manage client (D)

10.4 Example

Test case, more thorough variation

After every U all R-actions

- Manage client (C)
- Register membership (needed for next actions)
- Manage Client (R)
- Manage membership (R)
- Request information on seats (R)
- Manage Client (U)
- Manage Client (R)
- Manage membership (R)
- Request information on seats (R)
- Manage Membership (U)
- Manage Client (R)
- Manage membership (R)
- Request information on seats (R)
- Manage Client (D)
- Manage Client (R)
- Manage membership (R)
- Request information on seats (R)

Summary

- Description of CRUD operation
- Example of the CRUD operations on different entities
- Test Situations, Test Cases, Coverage
- Examples on different test situations



Add the notes here.

Review Question

- The following statement is true or false:
Consistency Check is a static test whereas
Completeness check is a dynamic test
 - True/False
- The overview of data life cycle is obtained using
 - CRUD Matrix
 - Flow Diagram
 - SRS
 - None of the above



Add the notes here.