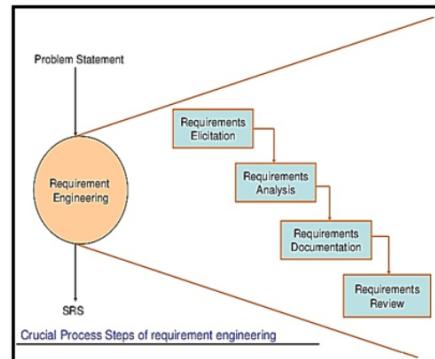


Requirement Engineering



Here you go Tom! You have the requirements for a Matrimonial website in hand. This is the problem statement.



Requirement Elicitation



Elicit requirements from all stakeholders

Address problems of scope

Address problems of understanding

Address problems of volatility (changing requirements)

Requirement Analysis



The process of establishing the services the system should provide and the constraints under which it must operate.

The process of studying and analyzing the customer and the user/stakeholder to arrive at a definition of software requirements.

Requirement Analysis



Why is Requirement Analysis difficult?

Different “worlds”

- Bridging the gap between the client and the software developer
- Knowing what should be done VS knowing what to let a computer do



Users/stakeholders are not a uniform group

- conflict between cost and usability / performance / features
- conflicting demands from different departments

Getting the good (ideal) system

Vs
possibility of building it well

Requirement Analysis



Other factors

- Expectations, the final solution is difficult to imagine by the users
- Scope of the system
 - needs well defined boundaries
- Current vs. future system
 - resistance to change
- Process of negotiation between users and designers

Requirement Analysis



Goals of requirement Analysis and specification phase

- Understand the user's requirements
- Remove inconsistencies, anomalies, etc. from requirements
- Document requirements properly in an SRS document

Requirement Analysis



A person who performs requirement analysis is called a system analyst

- Understands User requirement
- Collects data needed for the user requirement
- Writes the software requirement specification



Requirements analysis consists of two main activities:

- Requirements gathering
- Analysis of the gathered requirements



Requirement Analysis



Requirement gathering

- Observation of existing systems
- Studying existing procedures
- Discussion with the customer and the end-users
- Having questionnaires to understand user requirements
- Conducting discussion with domain experts to understand the system

Requirement gathering for automating system

- Analyst can easily obtain
 - ✓ Input and output formats
 - ✓ Accurate details of the operational procedures

Requirement Analysis



Analyzing the gathered requirement

- Understand the user requirements,
- Detect and remove inconsistencies, ambiguities, and incompleteness

Incompleteness and inconsistencies

Resolved through further discussions with the end-users and the customers

Incomplete Requirement



Some requirements have been omitted:

- Due to oversight.

Example:

- A solar heater specifies what it will do on a 'sunny day' but it fails to specify what it will do on a 'rainy day'
- The analyst has not recorded when temperature falls below 90 degrees.
- heater should be turned ON
- water shower turned OFF.

Requirement Analysis



Several things about the project should be clearly understood by the analyst

- What is the problem?
- Why is it important to solve the problem?
- What are the possible solutions to the problem?
- What complexities might arise while solving the problem?

Software Requirements Specification



Outcome of the analysis phase

Contains all the information about the requirements gathered

Main aim of requirements specification:

- Systematically organizes the requirements arrived during requirements analysis
- Documents requirements properly.

SRS



SRS document concentrates on

- What needs to be done
- Carefully avoids the solution ("how to do") aspects.

The SRS document is useful in various contexts

- Statement of user needs
- Contract document
- Reference document
- Definition for implementation

Properties of a good SRS



Concise

Specify what the system must do

Easy to change

Consistent

Complete

Traceable

Verifiable

SRS Document



SRS document normally contains three important parts

SRS

Functional requirements

Nonfunctional requirements

Constraints on the system.

Functional Requirement



Functional requirements describe

- A set of high-level requirements
- Each high-level requirement
- Takes in some data from the user
- Outputs some data to the user
- Might consist of a set of identifiable functions which process the input

Constraints



Constraints describe things that the system should or should not do.

- For example,
- Standards compliance
- How quickly the system can produce results so that it does not overload the other system to which it supplies data, etc.
- Hardware to be used
- Operating system
- DBMS to be used
- Capabilities of I/O devices
- Data representation

SRS Document



Organization of SRS Document

- Introduction
- Functional Requirements
- Nonfunctional Requirements
 - External interface requirements
 - Performance requirements
- Constraints

SRS Document Structure



1. Introduction	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Project Scope.....	1
1.5 References.....	1
2. Overall Description	2
2.1 Product Overview.....	2
2.2 Product Features.....	2
2.3 User Classes and Characteristics.....	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints.....	2
2.6 User Documentation.....	2
2.7 Assumptions and Dependencies.....	3
3. System Features	3
3.1 System Feature 1.....	3
3.2 System Feature 2 (and so on).....	4
4. External Interface Requirements.....	4
4.1 User Interfaces.....	4
4.2 Hardware Interfaces.....	4
4.3 Software Interfaces.....	4
4.4 Communications Interfaces.....	4
5. Other Nonfunctional Requirements	5
5.1 Performance Requirements.....	5
5.2 Safety Requirements.....	5
5.3 Security Requirements.....	5
5.4 Software Quality Attributes.....	5
6. Other Requirements	5
Appendix A: Glossary	5
Appendix B: Analysis Models	6
Appendix C: Issues List	6

ERD



Entity Relationship Diagram

ER diagram is widely used in database design

- Represents conceptual level of a database system
- Describes things and their relationships in high level

ERD



Entity

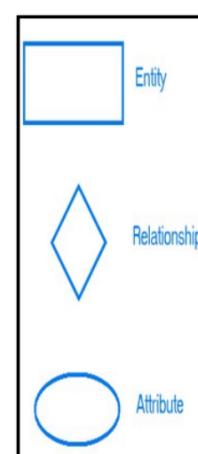
- An entity is a business object that represents a group, or a category of data.

Attribute

- Properties of an entity.

Relationship

- specifies the relations among entities



ERD



Relationship specifies association between two entities.

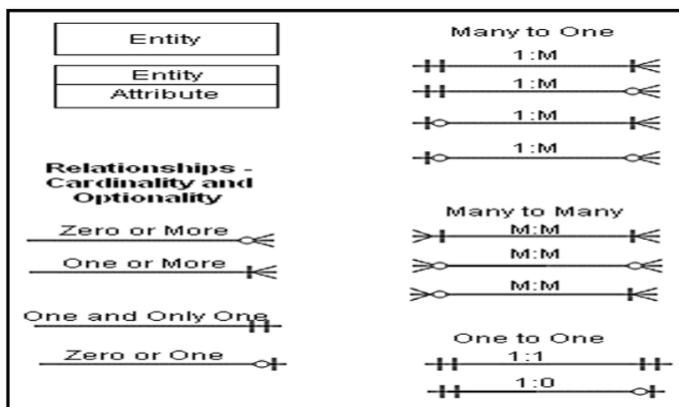
Cardinality

- One instance of an entity maps to how many instances of other entity
- Many-to-Many Relationships
- One-to-Many Relationships
- One-to-One Relationships
- Recursive Relationships

Optionality

- Is the relationship mandatory or optional
- Mandatory Relationships
- Optional Relationships

ERD Example



ERD Example

