# Case Studies:

## Insulin pump control system

* + 1. Customized
    2. Embedded
    3. Waterfall

## Mental Health Care-Patient Management System

1. Customized
2. Interactive transaction-based (Web)
3. Incremental/Prototype

## Wilderness weather station

1. Generic
2. Data collection
3. Waterfall

# Models:

## Spiral Model:

Provides support for Risk Handling. Each loop of the spiral is called a Phase of the software development process. The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks.

The Radius of the spiral at any point represents the expenses of the project so far, and the angular dimension represents the progress made so far in the current phase.

Diagram

Description automatically generated

Each phase of the Spiral Model is divided into four quadrants as shown in the above figure.

1. Objectives determination and identify alternative solutions: Requirements are gathered from the customers and the objectives are identified, elaborated, and analyzed at the start of every phase.
2. Identify and resolve Risks: During the second quadrant, all the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution are identified and the risks are resolved using the best possible strategy. At the end of this quadrant, the Prototype is built for the best possible solution.
3. Develop next version of the Product: During the third quadrant, the identified features are developed and verified through testing. At the end of the third quadrant, the next version of the software is available.
4. Review and plan for the next Phase: In the fourth quadrant, the Customers evaluate the so far developed version of the software. In the end, planning for the next phase is started.

### Advantages of Spiral Model:

* Risk Handling
* Good for large projects
* Flexibility in Requirements
* Customer Satisfaction

### Disadvantages of Spiral Model:

* Complex
* Expensive
* Difficulty in time management

## V-Model:

The V-model is a type of SDLC model where process executes in a sequential manner in V-shape. It is also known as Verification and Validation model. It is based on the association of a testing phase for each corresponding development stage. Development of each step directly associated with the testing phase. The next phase starts only after completion of the previous phase.

It is used when requirements are clearly defined and fixed and when sample technical resources are available with technical expertise.



Verification: It involves static analysis technique done without executing code. It is the process of evaluation of the product development phase to find whether specified requirements meet.

Validation: It involves dynamic analysis technique, testing done by executing code. Validation is the process to evaluate the software after the completion of the development phase to determine whether software meets the customer expectations and requirements.

### Design Phase:

* Requirement Analysis
* System Design
* Architectural Design
* Module Design

### Testing Phases:

* Unit Testing
* Integration testing
* System Testing
* User Acceptance Testing

### Principles of V-Model:

* Large to Small
* Data/Process Integrity
* Scalability
* Cross Referencing

### Advantages:

* This is a highly disciplined model and Phases are completed one at a time.
* V-Model is used for small projects where project requirements are clear.
* Simple and easy to understand and use.
* This model focuses on verification and validation activities early in the life cycle thereby enhancing the probability of building an error-free and good quality product.
* It enables project management to track progress accurately.

### Disadvantages:

* High risk and uncertainty.
* It is not a good for complex and object-oriented projects.
* It is not suitable for projects where requirements are not clear and contains high risk of changing.
* This model does not support iteration of phases.
* It does not easily handle concurrent events.

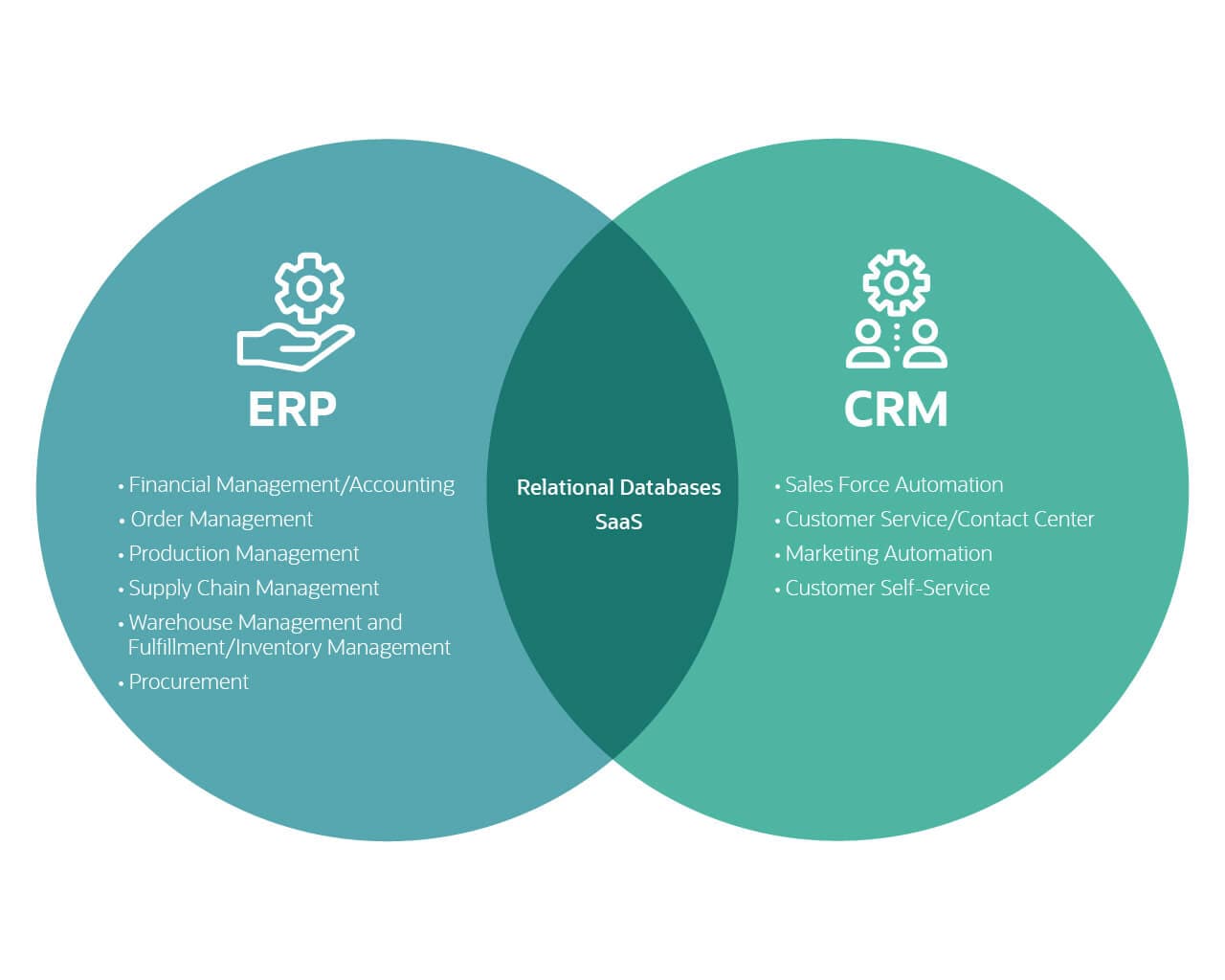
# CRM VS ERP:

## What is CRM?

CRM or customer relationship management is software that manages all the ways a customer interacts with a business.

## What is ERP?

Enterprise resource planning evolved out of material requirements planning (MRP), which was a way for manufacturers to understand and manage all the resources needed to operate a successful business. ERP serves as a shared database to all the parts of an organization. It also extends to inventory management, order management, supply chain management and data related to services organizations.



## What is the Difference Between CRM and ERP?

While the entire organization will come to rely on both ERP and CRM systems, the fundamental difference between ERP and CRM is that ERP is primarily for financial data and the finance department, while CRM is customer data used by the sales and customer service departments. The former is commonly referred to as the back office, and the latter is the front office.

Some ERP systems include a CRM component, while others do not, but CRM software systems do not include ERP components.