

# **Software Construction and Decision Making**

Alejandro Benítez

Class Notes

February 09, 2026

who knows what will be here

# Contents

1. Socioformadores .....	5
1.1. Mareé .....	5
1.2. Necesidad .....	5
1.3. Módulos .....	5
1.3.1. Menú .....	5
1.3.2. Administración .....	5
1.3.3. Experiencia .....	5
1.3.4. Diseño .....	6
1.3.5. Lealtad .....	6
1.3.6. Métricas .....	6
1.4. Preguntas .....	6
1.5. Notas .....	6
2. Project Management .....	7
2.1. An Introduction .....	7
2.1.1. Projects and Project Management .....	7
2.1.2. PMBOK .....	7
2.1.2.1. Project Integration Management .....	8
2.1.2.2. Project Communication Management .....	8
2.1.2.3. Project Scope Management .....	8
2.1.2.4. Project Schedule and Cost Management .....	8
2.1.2.5. Project Quality and Risk Management .....	8
2.1.2.6. Project Resource and Procurement Management .....	8
2.2. A Review on Objectives .....	8
2.3. Scope .....	9
2.3.1. Initiation .....	9
2.3.2. Scope Planning .....	10
2.4. Scope Definition .....	10
2.5. Work Breakdown Structures .....	10
2.6. Context Diagram .....	10
2.7. Project Charter .....	10
2.8. Addendum .....	11
3. Information Systems .....	12
3.1. An Introduction .....	12
3.1.1. Definitions .....	12
3.1.2. Architecture .....	12
3.2. DB & DBMS .....	13
3.2.1. DBMS .....	13
3.2.2. Objetivos .....	14
3.3. Notación Modelo Entidad Relación (MER) .....	14
3.3.1. Entidades .....	14
3.3.2. Asociaciones .....	14
3.3.3. Atributo .....	15
3.3.4. Metodología .....	15

3.4. MER Extendido .....	15
3.4.1. Roles .....	15
3.4.2. Relaciones ISA .....	15
3.4.3. Entidades fuertes y débiles .....	16
3.4.4. Restricciones de integridad adicionales .....	16
3.5. Traslado de MER a Tablas .....	16
3.5.1. Procedimiento de Transferencia .....	16
3.5.2. Procedimiento para MER Extendido .....	17

# 1. Socioformadores

## 1.1. Mareé

**Mauricio Hernández Siles** (442 443 5874), Joel, María. @mareecrepe

Crepería. Experiencia completa.

Proyecto en fortalecer la relación con su comunidad. Incrementar la lealtad que tienen.

- Sistema de royalty para clientes. Mobile. Tarjeta digital.
- Menú digital, personalizado. Horarios y disponibilidad. Futuro, pedir comida en la aplicación.
- Portal administrativo.
- Notificaciones whatsapp, programar mensajes.
- Adaptable y disponible en los teléfonos.

Quieren una solución integral personal. Evitar limitantes.

Apoyo con el diseño.

## 1.2. Necesidad

El objetivo principal es fortalecer la relación con su comunidad, e incrementar la lealtad de los consumidores. La experiencia del cliente es muy importante.

## 1.3. Módulos

### 1.3.1. Menú

Debe de ser muy customizable, dependiendo de lo que quiero y que le quiero poner el menú cambia. Yo puedo armar mi propia crepa. Incluso manejar toppings o ingredientes de temporada. Debe de haber dos modelos de menú de crepas

- Arma tu crepa
- Crepas predefinidas que pueden ser modificadas

Proveen café, crepas, y waffles.

### 1.3.2. Administración

Poder modificar precios, ingredientes, usuarios, de temporada, horarios de atención,

### 1.3.3. Experiencia

Primero era un modelo *to go*, pero no funcionaba tan bien. Ahora funciona como un restaurante, llegan, menus, comandan.

Si le gustaría poder educar a su cliente a poder ordenar por medio de una aplicación.

Son mayormente mujeres, y si son hombres son es porque vienen con su pareja. Es en un casona histórica de Mazatlán.

No quiere cambiar su tipo de venta. Hay un prepa cerca y muchos van, por lo que estaría bien que los chavos puedan utilizarlo. No se quiere alejar del cliente, se quiere acercar por medio de la aplicación.

Saber del cliente, saber su nombre, si se fue de viaje, tener ese acercamiento con el cliente y la aplicación debe poder complementar eso.

Pedidos caen a un sistema que hace un sonidito especial, y manualmente se ingresa a la comanda. Tener una ipad a full volumen para reconocer que es del sistema. Sería bien tener un pedido que sepa quien estaba recibiendo esas comandas para saber si hubo algún error.

#### **1.3.4. Diseño**

Girly, enfocado a la mujer. Pero, fácil de usar y amigable a los hombres.

#### **1.3.5. Lealtad**

Tienen una tarjetita que va puntuando. Llevar el control de saber que consume, invitarlo a volverlo a consumir. Se tiene que tener información para poder dar recomendaciones y mensajes estratégicos. Tiene que ser personalizado, por ejemplo descuentos en cumpleaños. En consumo de x cantidad puedes participar en rifas, etc.

#### **1.3.6. Métricas**

Poque no se mueven algunas crepas. De dinero no se necesita mucho. Se enfoca más en la visita del cliente, si se la pasó bien, etc.

### **1.4. Preguntas**

- No me queda muy claro a que se refieren con un portal administrativo.
- Que tipo de recompensa quieren ofrecer, puntos, descuentos, productos gratis
- La idea es que hayan membresias
- El menu cambia segun horario

### **1.5. Notas**

Va a haber muchas oportunides para conectar con terceros. Apple pay, google wallet, what-sapp.

## 2. Project Management

### 2.1. An Introduction

The course seeks to teach techniques and knowledge for the efficient project management. It is mainly based on the Project Management Institute (PMI) and its research.

First it is important to realize that few are the projects that are successful. Only about 17% of all projects can be considered successful. Even optimistic estimates only reach about 20 to 30%.

There are some points of suspicion, points which can give us insight into why projects fail so often. These are but some of them:

- Incomplete or changing specifications and requirements
- Lack of user involvement
- Not enough technical knowledge
- Inadequate use of methods and tools
- Unrealistic expectations

We can conclude the main reason is **weak project management**.

#### 2.1.1. Projects and Project Management

But first let us define what a project even is. A project is a temporal effort in order to bring a product, service or result to life. This definition implicates three things: effort, which is the use of organized resources; time constrainss, for it must be completed at some point; and the product itself, which must be alive, functioning.

Knowing this we might be able to answer, what is project management then?

---

Project management is the exercise of knowledge, abilities, tools and techniques in the activities that comprise a project in order to meet the project requirements.

---

#### 2.1.2. PMBOK

Now PMI, a non lucrative association, is the author and proprietary of the **PMBOK**, the most complete manual on project management. It establishes standards, it homogenizes training programs and even certifies individuals. And PMBOK is its Project Management Body of Knowledge, which, among other things, describes the environment wherein the projects will be developed, describes and organizes the project characteristics, describes the necessary knowledge to manage a project (divided into 9 areas), and establishes a shared vocabulary.

The PMBOK manages mainly 9 areas, namely:

- Project Integration Management
- Project Communication Management
- Project Scope Management
- Project Schedule Management

- Project Cost Management
- Project Quality Management
- Project Risk Management
- Project Resource Management
- Project Procurement Management

#### **2.1.2.1. Project Integration Management**

Includes the necessary processes to guarantee that the distinct elements of the project are well coordinated. It includes the development of the project plan, the execution of said plan, a control for the changes made. This is a recursive process. A plan is made, then executed and revised by the control. With the feedback from the control the plan is altered, then executed and so on. Once the control gives a green light the process is terminated.

#### **2.1.2.2. Project Communication Management**

It deals with how to generate, collect, distribute, store, recover and delete the projects information.

#### **2.1.2.3. Project Scope Management**

How to determine if the project contains and only contains the necessary work to execute the project. How to control the use of resources and execution of activities so that the project may have the exact desired characteristics and functionalities.

#### **2.1.2.4. Project Schedule and Cost Management**

How to determine the order and sequence of activities to deliver in time. How to assign resources so that the project stays within budget. How to use the resources, time and money.

#### **2.1.2.5. Project Quality and Risk Management**

Quality deals with how to satisfy the project necessities by the continued improvement of processes. Risk deals with how to identify events that are out of the project control, and how to act accordingly.

#### **2.1.2.6. Project Resource and Procurement Management**

Resource means how to organize and administrate the team for the project, balancing participation, decision taking and commitment. Procurement means how to acquire the goods and services required by the project, as well as knowing how to select from providers.

### **2.2. A Review on Objectives**

Any project must be divided in objectives. But first it is important to identify the necessity or problem, the **pain points** of the client. Only afterwards can the objectives be identified and detailed.

- Objective (usually in text)
- Functional (usecases diagrams)
- Non functional (quality attributes)
- Information (catalogues, fields, relations)
- Business rules

- Interface
  - User GUI
  - Communication (between other systems)

Extensible is a system that is able to grow by implementing more. Scalable is that the system grows just by inputting more computer power.

**Question:** How relevant is it nowadays the PMBOK and are big companies actually using it?

## 2.3. Scope

It is composed of all, and only those, necessary activities, works and processes that guarantee a successful project. It includes:

- Initiation
- Scope planning
- Scope definition
- Scope verification
- Scope change control

The scope is the aggregate of the *products* and *services* that need to be delivered, and defines what should be included but also what should not be included. These objectives are written in a SMART format.

But before diving into this subject, let's differentiate between *product scope* and *project scope*.

Product scope deals with the characteristics and functionalities that should be included in a given product or system. The success of it is verified with the requirements (functional, nonfunctional, business, etc.).

On the other hand, project scope deals with the work and tasks that need to be done in order to deliver the product with said characteristics and functionalities. Its success is verified with the project plan.

### 2.3.1. Initiation

Is the formal authorization of a new project or phase. It guarantees organizational support for the project. Among other things it defines the high level objectives, secures the necessary resources and approvals, confirms the alignment of the project with the strategic objectives, and assigns a project boss.

Since there may be many project ideas, only the best one should be considered. Therefore a method of prioritization is necessary. The available methods include: focusing on general necessities, categorizing the projects, financial analysis, and punctuation methods.

Once the project has been selected it must be formalized so that everyone involved in the project is on the same page. This formalization must contain:

- Objectives
- Success criteria
- Restrictions
- Assumptions

- Roles and responsibilities

### **Is this formalization the same as the project charter or a separate document?**

There must be a project charter that recognizes the project and informs about the necessities of the project and the description of the product. This charter must have authority, and therefore must be signed by a high ranking official (xd).

#### **2.3.2. Scope Planning**

The development of a written scope statement that will serve as the foundation for future decisions regarding the project. It also establishes an agreement between the project and client in order to identify both the objectives as well as the things that will be delivered.

This statement must include (a) a project justification, (b) the project product, (c) the things to be delivered, and (d) the project objectives (written in a SMART method).

The scope management plan must describe how the project will be managed and how scope changes may be integrated. It is part of the global project plan. It must include the stability expectations and a clear description as to how scope changes will be identified and classified.

#### **2.4. Scope Definition**

Now the work must be divided into several more manageable pieces. Here we seek to

- Improve the precision of estimates for time, cost, and resources.
- Define the baseline by which the project will be compared and controlled.
- Help clearly define the responsibilities of each task.

A deliverable is every product, result or measurable element, tangible and verifiable that will be delivered.

#### **2.5. Work Breakdown Structures**

no idea what this is

It is the process of dividing the deliverables and work in smaller and more manageable products. It is a hierarchical decomposition delivery oriented.

A work package is the lowest level. We now our division has gotten there when we can (a) state when it begins and ends, (b) estimate costs and efforts, and (c) monitor and control it.

The detail level must be enough.

#### **2.6. Context Diagram**

It helps us identifying the scope of the project, the people interested and how data flows through the system.

#### **2.7. Project Charter**

Must include a description of the project, the people responsible, budget and time, what should be in the project and what not.

There must also be milestones that help divide the project and get tangible results, as well as knowing exactly how bad you are developing.

## 2.8. Addendum

- The people **interested** in the project are those who benefit and are affected by it.
- **Project agreement** is the document that formalises what the goals, scope and overall things that will be done in the project, which includes the time. It must also outline what the project will contain and what it will not.

### 3. Information Systems

#### 3.1. An Introduction

An information system is defined as a set of interrelated components that compile, process, store and distribute information with the goal of decision taking and control over organization.

##### 3.1.1. Definitions

- **Data:** Is the symbolic representation of an attribute or characteristic of an entity. It does not have semantic meaning but may be conveniently processed into a more valuable thing.
- **Information:** Is an organized set of data that has been processed and thus has semantic meaning.
- **Input:** Is the information (data?) produced by the user. An input also can be produced by networks, etc.
- **Processing:** Are the actions that take an entry and generate useful information for the user.
- **Output:** Is the information that the computer will show to the user.
- **Feedback:** Is the added value that allows users to take actions.

##### 3.1.2. Architecture

It is divided in mainly five levels (types), each one with a specific user in mind.

1. **Transaction Processing System (TPS):** Help in the daily activities of a enterprise. They are the heart of the information system. It comprises the data collection, data manipulation, storage, and report production.

Its goal is to process data, be exact and correct, maintain data integrity, produce opportune reports and documents, and overall improve laboral efficiency. It, thus, has a big volume I/O, needs high storage capabilities, and impacts a high number of user (specially when there is an error and everyone is stroke).

A transaction means sort of CRUD, create, recover, update and delete information. There are three processing types:

- In line: In the moment as it arrives, though it does not matter if it lags behind (register user).
- In batch: Waits a bit to gather up and then processes.
- Realtime: Processes instantly as it arrives, where the time is critical (air traffic).

2. **Office Automation Systems (OAS):** Provide help for daily activities to increase employee efficiency of those who work with the information (e.g. word processors, email, schedules, etc.).

**Knowledge Work Systems (KWS):** Provide help for specialized employees in the creation and integration of new knowledge for the institution, for instance software that aids architects in blueprint design.

3. **Management Information Systems (MIS):** A collection of people, procedures, databases, and devices needed to display information to the managers and decision takers so that they may achieve the objectives. They seek to show a general vision of how things look and how and which operations take place.

These systems produce reports, which may be scheduled, demanded or exceptional (when a critical situation creates the need).

**Decision Support Systems:** It is similar to a MIS, but its purpose is to take decisions regarding specific problems, semi-structured problems or non-structured ones. It also gathers information from many places, but it must also be able to perform complex and sophisticated analysis using statistical packages. It must also be visual. A DSS serves as a *what if* visualizer, allowing the user to see the impact of hypothetical cases.

4. **Group Decision Support Systems (GDSS):** This system aids group decision taking. Its design is special, as it must encourage creative thinking and effective communication. It must also be easy to learn and use, while also being flexible enough to accommodate for the different kinds of people. It should also have some degree of anonymity in order for to avoid source bias.

**Expert Systems (ES):** A system that acts as a human expert in certain field or area. It helps identifying issues, predicting future events and aiding in the design of new products.

5. **Executive Information Systems (EIS):** Encompasses all critical information that influences decision taking. It aids non structured decision taking with KPIs graphicly. It uses quality information, reduced use of the keyboard, secure access, fast response time, remote access and perfect fitted design.

## 3.2. DB & DBMS

Se puede pensar en una base de datos como una colección de archivos interrelacionados lógicamente, cuyos datos han sido cuidadosamente diseñados para que estos archivos contengan la menor repetición de datos y hechos.

El uso de una base de datos es muy conveniente ya que nos permite trabajar con una mayor cantidad de datos, además de que se le puede dar una estructura bien diseñada para facilitar su uso.

### 3.2.1. DBMS

Un sistema de gestión de base de datos consiste en una colección de datos y un conjunto de programas para controlar esos datos. A la colección de datos interrelacionados se le denomina como la base de datos, y al conjunto de programas como el DBMS, que crea un entorno conveniente y eficiente para extraer y almacenar información de la base de datos.

### **3.2.2. Objetivos**

- Eliminar la redundancia y la inconsistencia de los datos. El hecho de tener redundancia solo aumenta los costos de almacenamiento y acceso. Además, puede llevar a una inconsistencia de datos donde los diferentes registros contienen diferente información y no concuerden entre sí.
- Elimina la dificultad de acceso a la información. Las bases de datos proporcionan un sistema conveniente y eficiente para recuperar información, pero que a la vez es lo suficientemente generales para poder ser utilizado en varios lugares.
- Los datos están naturalmente relacionados.
- Seguridad al acceder de manera concurrente.
- Seguridad para evitar que cualquiera pueda acceder a la información.
- Integridad de los datos. Se pueden añadir restricciones a los datos dentro de la base de datos para asegurar que ciertas reglas se cumplan.
- Crear copias y backups de la base de datos.:.

### **3.3. Notación Modelo Entidad Relación (MER)**

Es un método y diagrama que nos permite diseñar bases de datos de una buena manera. Un modelo entidad relación se compone de ciertos elemetos, los cuales son explicados a continuación.

#### **3.3.1. Entidades**

Representan el objeto básico de nuestro diagrama, y se refieren a cualquier objeto del mundo real sobre el cual queremos tener información. Se representa como un rectángulo.

#### **3.3.2. Asociaciones**

Representan la interrelación que existe entre las entidades del modelo.

Se debe de procurar que los nombres de las asociaciones sean precisos, usualmente un solo verbo que describe la relación entre las entidades involucradas. Una asociación representa solo una forma en la que las entidades se relacionan, pero puede que dos entidades se relacionen de varias formas, por lo que de debe de especificar las distintas asociaciones que hay entre esas entidades.

Adicionalmente, una asociación puede tener atributos descriptivos, lo que nos permite almacenar información extra acerca de el tipo de relación que se tiene.

El grado de una asociación se refiere a la cantidad de entidades que participan en la relación. Esto también quiere decir que se pueden tener varias entidades participando en una sola asociación. Sin embargo es aconsejable restringirlo a solo relaciones binarias.

La cardinalidad es el número de elementos que participan en una asociación. Existen tres casos generales: asociaciones uno a uno (1:1), uno a muchos (1:N), y relaciones muchos a muchos (N:M). Para casos más específicos se puede establecer explícitamente el valor de N.

Existen dos tipos de participación en una asociación. Primero se encuentra la parcial u opcional. Dentro de esta categoría no todas las instancias de una entidad tienen que estar relacionadas con la otra entidad. Se representan con una sola línea. Por otra parte, las asociaciones obligatorias o parciales deben de estar siempre relacionadas. Se dice que existe una participación total. Se representan con dos líneas.

### 3.3.3. Atributo

Los atributos son la información que nos interesa recolectar. Son características y valores. Se le pueden representar como elipses y como una lista de los nombres de los atributos.

El conjunto de valores que puede tomar un atributo se le llama dominio.

Existen varios tipos de atributos:

- **Simples o compuestos:** Los compuestos están formados por varios atributos, los almacenados no.
- **Monovaluados o multivariados:** Los mono solo pueden almacenar un valor, los multi varios valores.
- **Almacenados y derivados:** Los almacenados son almacenados. Los derivados no son almacenador, sino que se pueden obtener mediante cálculos y funciones sobre otros atributos que sí son almacenados.

Un atributo muy importante y especial es el identificador de identidad. Puede ser uno o más atributos que identifican a una entidad como única.

### 3.3.4. Metodología

1. Identificar entidades
2. Incorporar atributos
3. Determinar identificadores
4. Identificar asociaciones
5. Determinar cardinalidad
6. Incorporar atributos en las asociaciones
7. Verificar con los requerimientos

## 3.4. MER Extendido

Estos son elementos que buscan expandir y complementar el modelo relación entidad. Sin embargo se debe de tomar en cuenta que estos elementos no se deben de abusar, deberían de considerarse como excepciones y no siempre deben de estar presentes, muchas veces el modelo no extendido es suficiente.

### 3.4.1. Roles

Se utilizan cuando una entidad juega más de un rol en una asociación, ya sea consigo misma (recursiva) o con múltiples entidades. Se representan como un rectángulo pegado a la entidad.

### 3.4.2. Relaciones ISA

Utilizan entidades generalizadoras (superclase), y entidades especializadoras (subclases).

- Una **superclase** es todo tipo de entidad a la que se le definen subclases.

- Una **subclase** es una entidad que cuenta con todos los atributos de la superclase, pero adicionalmente cuenta con atributos extra.

Estas son entidades distintas, por lo que un objeto no puede jugar el rol de ambas, a diferencia de un rol donde esto sí sucede. La forma en la que estas entidades operan es muy similar a la herencia en el paradigma de programación orientada a objetos.

### 3.4.3. Entidades fuertes y débiles

- **Fuertes:** Son las entidades normales. Existen por sí mismas.
- **Débiles:** Son las entidades que dependen de otras. Si la entidad en la que dependen desaparece, entonces la entidad débil también lo hace. Se identifican con un doble rectángulo.

### 3.4.4. Restricciones de integridad adicionales

- **Cotas de cardinalidad:** Son una extensión de la notación de cardinalidad. Se añade información acerca del número mínimo y máximo que pueden haber de asociaciones. Se delimitan de la forma `min:max`.
- **Entidad atributo:** Son restricciones adicionales que existen entre los atributos de una entidad. Algunos ejemplos de estas son restricciones de recursos humanos, ecológicas, de tiempo, instalaciones, transporte, recursos financieros, materiales, etc.

¿Cómo se especifican estos dentro del MER?

## 3.5. Traslado de MER a Tablas

Para poder llevar nuestro modelo a la realidad es necesario hacer una traducción del modelo entidad relación a algo más concreto, que en este caso llamaremos modelo relacional. Tal modelo se definirá de la siguiente manera:

```
nombre(columna1, columna2, columna3, ...)
```

Para reconocer la llave primaria de la tabla esta se subrayará, mostrando que la columna o conjunto de columnas representan una llave.

### 3.5.1. Procedimiento de Transferencia

Para cada entidad se define una **tabla** cuyas **columnas** corresponden uno a uno a los atributos de la entidad. De no existir un identificador en la tabla que garantice unicidad entonces se debe de agregar.

Para las asociaciones con **cardinalidad 1:1** solo se tiene que agregar la llave primaria de una tabla en la otra. Es indistinto cual llave es agregada.

Para las asociaciones con **cardinalidad 1:N**, se debe de agregar la llave primaria de la entidad que ocurre varias veces en la entidad que solo ocurre una vez. Por ejemplo, si un profesor puede dar varios grupos pero solo uno puede dar un grupo, entonces la llave primaria del profesor debería de estar en cada grupo.

Finalmente, para **cardinalidades de tipo N:N** se debe de crear una nueva tabla donde se encuentran las dos llaves primarias de ambas entidades, además de los atributos extra que se requieran. La llave primaria de esta tabla sería la concatenación de las llaves primarias de ambas entidades.

### 3.5.2. Procedimiento para MER Extendido

Las entidades con **relaciones ISA** son 1:1, sin embargo su mapeo es un caso particular de las relaciones 1:N. La llave primaria se encuentra en la entidad generalizadora (superclase o clase padre), y las entidades especializadoras (subclase o clase hija) simplemente comparten exactamente la misma llave primaria. Se dice que las entidades generalizadoras heredan su llave primaria a las entidades especializadoras.

Para el caso de las **entidades débiles**, la entidad fuerte de la que dependen le hereda su llave primaria, similar al caso anterior. Sin embargo, tal entidad débil ya tiene un identificador único, por lo que su llave primaria se vuelve la concatenación de la llave primaria heredada con el identificador original.

Finalmente, el caso de los **roles** es muy similar. Secillamente se considera que es la misma entidad independiente del rol, y dentro de la llave foránea (que sería la llave primaria de una entidad igual) se especifica el rol que tal entidad foránea juega. Para asociaciones de N:N se sigue el mismo procedimiento, en donde se crea una tabla extra.