

Diseño de Software

Andrés Felipe Rentería Velandia, Andrés Leonardo Arias Uribe

2 de abril de 2019

Índice general

I	PROYECTO	7
1.	Caso de Estudio	9
1.1.	Introducción	9
1.2.	Descripción del problema	9
1.3.	Objetivo General	10
1.3.1.	Objetivos específicos	10
1.4.	Alcances	10
1.5.	Limitaciones	11
2.	Metodología	13
2.1.	Introducción	13
2.2.	Proceso de Software	14
II	DISEÑO	15
3.	Requerimientos	17
3.1.	Introducción	17
4.	Interacción	19
4.1.	Introducción	19
5.	Clases	21
5.1.	Introducción	21
6.	Patrones	23
6.1.	Introducción	23
6.2.	Prototipo	24

7. Estados	25
7.1. Introducción	25
8. Componentes	27
8.1. Introducción	27
9. Nodos	29
9.1. Introducción	29
10. Actividades	31
10.1. Introducción	31
 III REFLEXIONES	 33
11. Conclusiones	35
11.1. Introducción	35

Índice de figuras

2.1. Proceso RUP [4, 3, 2, 1]	14
6.1. Prototipo	24

Parte I

PROYECTO

Capítulo 1

Caso de Estudio

1.1. Introducción

En los últimos cincuenta años se ha desarrollado una revolución tecnológica de una manera tan marcada que ha obligado al mundo a adaptarse y evolucionar a un ritmo vertiginoso; este cambio es algo que afecta a todos los miembros activos de la sociedad, en especial a las empresas, pues han visto como su funcionamiento es cada vez más complejo, puesto que cada vez están en contacto con un número mayor de clientela, lo cual las obliga a buscar ser más productivas e implementan una mayor cantidad de personal. Por todo esto, en los últimos años las empresas han empezado a buscar soluciones tecnológicas que permitan simplificar los procesos que se desarrollan dentro de ellas y lograr tal objetivo de productividad.

1.2. Descripción del problema

Una empresa de cines, ubicada en Bogotá, busca desarrollar un software que le permita administrar su funcionamiento. Dicho software debe estar en la capacidad de gestionar la boletería de todos los cinemas que tiene a lo largo de la ciudad para cada una de las diversas funciones ofrecidas en las distintas horas del día en tareas como compra y reserva de boletas, al igual que las ventas de confitería de cada cinema. Además estará en la capacidad de llevar un registro y monitoreo de los empleados y el manejo de insumos y maquinaria de confitería que estén a disposición de cada cinema.

Durante el siguiente libro se realizará un estudio del problema planteado, buscando encontrar una solución eficiente mediante la implementación de la ingeniería de software.

1.3. Objetivo General

Generar una solución de software completa que permita administrar y monitorear las actividades principales de una cadena de cines, mediante el uso de metodologías y fundamentos básicos de ingeniería de software.

1.3.1. Objetivos específicos

1. Gestionar el manejo de boletería y funciones para cada uno de los cines a lo largo de la ciudad de Bogotá D.C. incluyendo tanto la compra y reserva de boletas como selección de asientos de las funciones.
2. Administrar el manejo de los insumos en la sección de confitería para cada uno de los cinemas de la ciudad, mediante el uso de bases de datos.
3. Desarrollar un cliente de uso para el público general y uno diferente de uso para personal administrativo.
4. Implementar un registro de los empleados que tiene la empresa en cada uno de los cinemas.

1.4. Alcances

Se implementará un cliente de uso para público general que podrá realizar las siguientes actividades:

- Compra y reserva de boletas.
- Selección de asientos.
- Visualización de películas en cartelera y próximos estrenos.
- Gestión de suscripción especial.
- Exposición y compra de confitería.
- Generación de tickets de pago.

Se implementará un cliente de uso para personal administrativo que podrá realizar las siguientes actividades:

- Gestionamiento de cartelera, próximos estrenos y confitería.

- Asignación de funciones.
- Manejo del inventario de confiteria.
- Gestionamiento de personal a nivel de registro.

1.5. Limitaciones

- El cliente público se limitará a la emisión de tickets de pago, no contará con la posibilidad de un módulo de pago.
- El cliente de personal administrativo no contará con el manejo de nómina, solamente se restringirá al registro de personal de trabajo.

Capítulo 2

Metodología

2.1. Introducción

contenido...

2.2. Proceso de Software

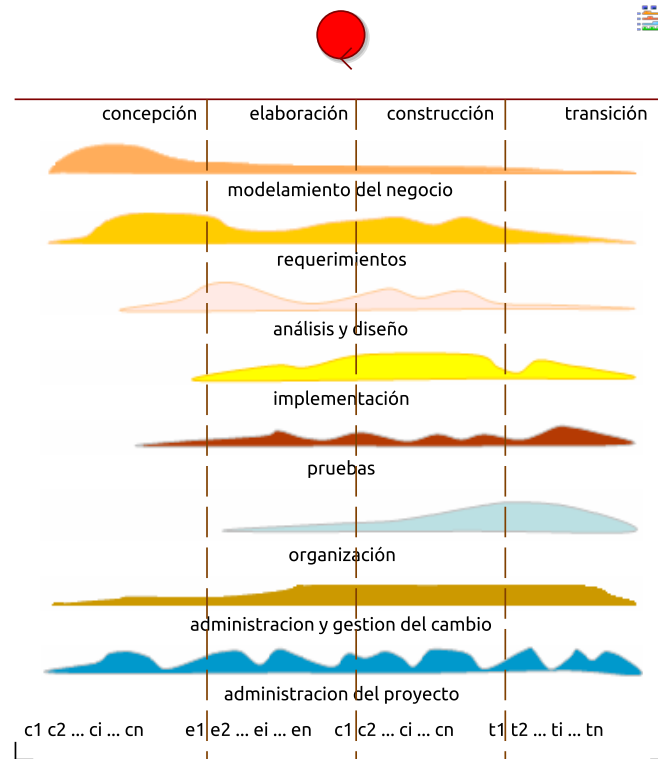


Figura 2.1: Proceso RUP [4, 3, 2, 1]

Parte II

DISEÑO

Capítulo 3

Requerimientos

3.1. Introducción

contenido...

Capítulo 4

Interacción

4.1. Introducción

contenido...

Capítulo 5

Clases

5.1. Introducción

contenido...

Capítulo 6

Patrones

6.1. Introducción

contenido...

6.2. Prototipo

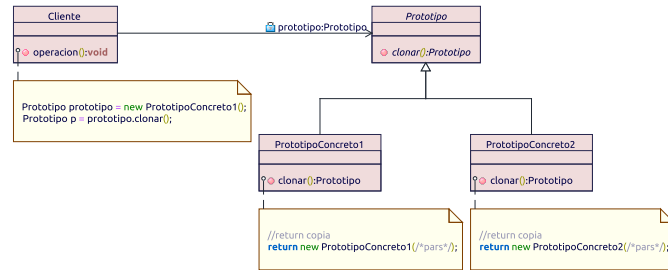


Figura 6.1: Prototipo

Capítulo 7

Estados

7.1. Introducción

contenido...

Capítulo 8

Componentes

8.1. Introducción

contenido...

Capítulo 9

Nodos

9.1. Introducción

contenido...

Capítulo 10

Actividades

10.1. Introducción

contenido...

Parte III

REFLEXIONES

Capítulo 11

Conclusiones

11.1. Introducción

contenido...

Bibliografía

- [1] D. Arcelli, V. Cortellessa, and D. Di Pompeo. Performance-driven software architecture refactoring. In *2018 IEEE International Conference on Software Architecture Companion (ICSA-C)*, pages 2–3, April 2018.
- [2] M. A. Babar, I. Gorton, and R. Jeffery. Capturing and using software architecture knowledge for architecture-based software development. In *Fifth International Conference on Quality Software (QSIC'05)*, pages 169–176, Sep. 2005.
- [3] L. Bass, P. Clements, R. Kazman, and M. Klein. Evaluating the software architecture competence of organizations. In *Seventh Working IEEE/IFIP Conference on Software Architecture (WICSA 2008)*, pages 249–252, Feb 2008.
- [4] Bolanos S. *Metaroceso de Software*.