|  |
| --- |
|  |

Software Architecture Document (SAD)

**Proyecto: Actividades Municipalidad de Limón**

**Revisión: 1.0**

**Estudiantes:**

**Froilán Moya Robles**

**María José González Araya**

# Historial de versiones

A continuación se presenta el historial de las revisiones a lo largo de la ejecución de este documento. **Ver Tabla 1.**

**Tabla 1.** *Historial de Revisiones.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Descripción** | **Autor** |
| 16/12/2018 | 1.0 | Versión inicial | María José González Araya |
|  |  |  |  |
|  |  |  |  |

Tabla de Contenidos

[**Historial de versiones**](#_tbjpbcrvcpi) **2**

[**Introducción**](#_r47f8ov5diq) **4**

[Propósito](#_qyb77pieigky) 4

[Alcance](#_q21iiawxzc4a) 4

[Definiciones, acrónimos y abreviaturas](#_2nfauyiya6ix) 4

[Referencias](#_ucpfwep34vfd) 4

[Visión general](#_v71svlr086sp) 4

[**Metas y restricciones arquitectónicas**](#_pvy9rz51pfr7) **4**

[**Representación arquitectónica**](#_2s8eyo1) **5**

[Vistas arquitectónicas](#_oj6iasgd1mth) 5

[Architectural Design Patterns](#_kr3xrj92hhzy) 5

[Architectural Style](#_pgrp24cbnpld) 6

[Architectural Process](#_75f5szdip8tx) 6

[**Architectural View Decomposition**](#_44sinio) **7**

[Use-Case View](#_mhz28fjgez74) 7

[Use-Case Realizations](#_t3pkw0e2b1s7) 8

[Architecturally Significant Use Cases](#_3nhj69u1ism0) 8

[Formulate Alternatives](#_xzxgi0uiqcqx) 9

[Evaluate Alternatives](#_rqmz2ufe28w) 9

[Make Decision](#_7159u1qvwff3) 9

[Design View](#_djwj1pksclvu) 9

[Overview](#_v6dzvvk6gizf) 9

[Process View](#_v0y3pty9tufa) 9

[Component View](#_queeja7mvm66) 9

[Overview](#_2v9frxfybq26) 10

[Deployment View](#_nkvj1yhqfsh) 10

[**Size and Performance**](#_9opbuu516546) **10**

[**Quality**](#_gvfppj5glwn9) **10**

[**Bibliography**](#_gtmn6gsm3z8) **10**

# Introducción

Esta introducción proporciona una descripción general de todo el documento de arquitectura de software para el sistema de soporte de planificación de la conservación. Incluye el propósito, el alcance, las definiciones, las siglas, las abreviaturas, las referencias y la descripción general del sistema.

## Propósito

Este documento proporciona una descripción general de la arquitectura del sistema de gestión de Actividades de la

Municipalidad de Limón.

El propósito principal de este sistema es brindarle a la municipalidad de Limón una herramienta la cual puedan

utilizar para tener una mejor cercanía con la comunidad limonense para poder comunicarles las actividades

realizadas por la municipalidad y de poder realizar sugerencias sobre el funcionamiento de la municipalidad y sus

actividades.

Este documento pretende capturar y transmitir las importantes decisiones arquitectónicas que se han tomado al

diseñar y construir el sistema. Es una forma mediante la cual los diferentes involucrados en el desarrollo del sistema

pueden comprender mejor los problemas a resolver y cómo se representarán con este sistema.

## Alcance

El alcance de este documento es la implementación de los documentos .

## Definiciones, acrónimos y abreviaturas

.

## Referencias

.

## Visión general

Este documento consta de 7 secciones, que se describen a continuación:

* La Sección 1 es simplemente una introducción a la arquitectura del software.
* La Sección 2 aborda los objetivos y limitaciones de la arquitectura del sistema.
* La sección 3 describe la representación arquitectónica del sistema.
* La Sección 4 describe las cinco vistas en las que se divide la documentación del sistema siguiendo el Proceso Unificado de Rational (RUP). Una representación visual de estas cinco vistas se puede ver en el Diagrama de desarrollo del sistema de software.
* La sección 5 de este documento habla sobre otras consideraciones del sistema, como el tamaño y el rendimiento del sistema.
* La sección 6 describe algunos problemas de calidad del sistema.
* La sección 7 es una bibliografía de las referencias utilizadas para crear este documento.

# Metas y restricciones arquitectónicas

La arquitectura ha sido diseñada con los siguientes objetivos en mente:

Las principales limitaciones de diseño e implementación del sistema son:

Sencillez

Flexibilidad

Una lista completa de los requisitos del sistema se puede ver en el documento de requerimientos ERS..

# Representación arquitectónica

A continuación se enlistan las diferentes representaciones arquitectónicas

## Vistas arquitectónicas

Al modelar, implementar y documentar un sistema se requiere que el sistema se vea desde diferentes perspectivas. Debido a esto, la arquitectura del sistema se representará en un enfoque de cinco vistas: Vista de caso de uso, Vista de diseño, Vista de proceso, Vista de componente y Vista de implementación. La siguiente es una breve descripción de cada una de las vistas. **Ver Figura 1.**



**Figura 1.** *Vistas arquitectonicas.*

**Use Case View:** El propósito principal de la vista de caso (use case view) de uso es definir los principales impulsores del sistema, que son los requisitos del sistema.

**Design View:** Esta vista contiene todas las definiciones del sistema, así como los diagramas de clase y objeto que representan los servicios que el sistema proporcionará a sus usuarios finales.

**Process View:** Esta vista mostrará los procesos que forman el mecanismo de los sistemas. Estos serán representados como diagramas de colaboración, secuencia y actividad.

**Component View**: Esta vista incluirá especificaciones del sistema y de la interfaz de usuario, es decir, los diferentes componentes que conforman el sistema.

**Deployment View:** Esta vista mostrará cómo los nodos de hardware de los diferentes sistemas cobrarán vida juntos, así como la forma en que se instalarán y desplegarán cada uno de los nodos de hardware.

Como se ve en la **Figura 1**, los requisitos, o la vista del caso de uso, son la fuerza impulsora principal de

este y cualquier sistema de arquitectura de software..

## Patrones de diseño arquitectónico

The design pattern used to create the CPSS is the MVC (Model View Controller) design pattern. The MVC design patter clearly separated the web application’s behavior, presentation and control. The modularity of this design pattern allows for easier code reuse, more centralized control, bugs easier to track down and code easier to modify. The presentation, or view, of the CPSS has been implemented keeping in mind the Model 2 usage pattern, which makes use of servlets as front controllers and maps incoming request to specific operation and selects views based on the model and session state. In some steps of the application the Model 1 pattern, in which a servlets is not used as the front controller, is also used. Some of the sections of the CPSS where Model 1 architecture is used are the DB Input For and the CPPE tool. These tools are very simple as far as navigation and a front controller is not needed since the controller is implied through the higher level container (the CPSS).

## Estilo arquitectónico

As with any other things, a style may be used to satisfy any functional, non-functional or aesthetic needs in a software system. The CPSS, in particular, follows the three-tier architectural style: presentation tier, business tier, and data tier. The following is a simple description of what will be included in each of the tiers:





* Presentation Tier: used to format and present the information to the user.
* Business Tier: used to implement the logic that will drive the system and the reason why the system exists.
* Data Tier: in charge of storing the data (databases) and other external services that the system may use.

## Proceso arquitectónico

El sistema sigue el Proceso Unificado de Rational (RUP), cuyo objetivo es permitir la producción del software de la más alta calidad que satisfaga las necesidades del usuario final dentro de un calendario y presupuesto predecibles. El RUP es un proceso iterativo dividido en cuatro fases: inicio (establecimiento del caso de negocios del proyecto), elaboración (establecimiento del plan del proyecto y arquitectura del sistema), construcción (implementación del sistema) y transición (implementación del sistema).

# Descomposición de la vista arquitectónica

A continuación se presenta los diferentes diagramas de los casos que representan las vistas arquitectónicas establecidas en el punto 3.1

## **Use-Case View**

La funcionalidad general del sistema se puede ver en el siguiente diagrama. **Ver Figura**

#### **Architecturally Significant Use Cases** Los casos de uso significativos desde el punto de vista arquitectónico son aquellos que “ejercitan” las partes más críticas de la arquitectura del sistema y demuestran la funcionalidad central del sistema.

## **Design View**

Esta sección describe la estructura lógica del sistema, así como cualquier definición o acrónimo. Comienza a partir de la descripción general de la arquitectura y luego presenta sus elementos estructurales y de comportamiento clave, como el uso y la dependencia. **Ver Figura**

## **Process View**

La vista del proceso describe la descomposición del sistema, así como las formas de comunicación entre procesos, como el paso de mensajes, la actividad entre componentes y la secuencia de mensajes. **Ver Figura**

* [Collaboration Diagram.doc](about:blank)
* [Sequence Diagram.doc](about:blank)
* [Activity Diagram.doc](about:blank)

## **Component View**

La vista de componentes describirá la organización global de componentes y subsistemas. **Ver Figura**

* [User Interface Specifications.doc](about:blank)
* [System Interface Specifications.doc](about:blank)
* Software Architecture Documentation.doc (this document)

## **Deployment View**