<ADMIN PI>

Software Development Plan

Version <1.0>

*[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]*

*[To customize automatic fields in Microsoft Word (which display a gray background when selected), select File>Properties and replace the Title, Subject and Company fields with the appropriate information for this document. After closing the dialog, automatic fields may be updated throughout the document by selecting Edit>Select All (or Ctrl-A) and pressing F9, or simply click on the field and press F9. This must be done separately for Headers and Footers. Alt-F9 will toggle between displaying the field names and the field contents. See Word help for more information on working with fields.]*

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <14/05/2019> | <1.0> | Versión preliminar como propuesta de desarrollo. | Adan Oswaldo Piceno  Marlene Tapia |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 5

1.1 Purpose 5

1.2 Scope 5

1.3 Definitions, Acronyms, and Abbreviations 5

1.4 References 5

1.5 Overview 6

2. Project Overview 6

2.1 Project Purpose, Scope, and Objectives 6

2.2 Assumptions and Constraints 6

2.3 Project Deliverables 6

2.4 Evolution of the Software Development Plan 6

3. Project Organization 6

3.1 Organizational Structure 6

3.2 External Interfaces 6

3.3 Roles and Responsibilities 6

4. Management Process 6

4.1 Project Estimates 6

4.2 Project Plan 7

4.2.1 Phase Plan 7

4.2.2 Iteration Objectives 7

4.2.3 Releases 7

4.2.4 Project Schedule 7

4.2.5 Project Resourcing 7

4.2.6 Budget 7

4.3 Iteration Plans 7

4.4 Project Monitoring and Control 7

4.4.1 Requirements Management Plan 7

4.4.2 Schedule Control Plan 7

4.4.3 Budget Control Plan 8

4.4.4 Quality Control Plan 8

4.4.5 Reporting Plan 8

4.4.6 Measurement Plan 8

4.5 Risk Management Plan 8

4.6 Close-out Plan 8

5. Technical Process Plans 8

5.1 Development Case 8

5.2 Methods, Tools, and Techniques 8

5.3 Infrastructure Plan 8

5.4 Product Acceptance Plan 8

6. Supporting Process Plans 8

6.1 Configuration Management Plan 8

6.2 Evaluation Plan 9

6.3 Documentation Plan 9

6.4 Quality Assurance Plan 9

6.5 Problem Resolution Plan 9

6.6 Subcontractor Management Plan 9

6.7 Process Improvement Plan 9

7. Additional Plans 9

8. Annexes 9

9. Index 9

Software Development Plan

# Introduction

Este Plan de Desarrollo del Software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto de creación de un sistema enfocado a nóminas ADMIN PI. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto ha sido basado en una metodología de Rational Unified Process (RUP), Es importante destacar esto puesto que utilizaremos la terminología de dicha metodología en este documento. Se incluirá el modelo del negocio y el alcance del proyecto. Se identifican todos los actores y casos de uso. Se desarrolla un plan de negocio para determinar qué recursos deben ser asignados al proyecto.

El enfoque desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

## Purpose

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

•El jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.

•Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

## Scope

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo del “Admin PI”. El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones. Para la versión 0.1del Plan de Desarrollo del Software, nos hemos basado en la captura de requisitos, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”, el cual se utilizará para refinar este documento. Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

## Definitions, Acronyms, and Abbreviations

***NO APLICA***

## References

* *Requirements Management Plan*
* *Configuration Management Plan*

## Overview

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto..

Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

Infraestructura — proporciona los requerimientos necesarios para poder implementar la solución propuesta. Son los requisitos para que el sistema funcione adecuadamente.

# Project Overview

## Project Purpose, Scope, and Objectives

***PURPOSE***

El propósito es poder mejorar el control de pagos dentro de cualquier establecimiento específicamente las facturaciones a nómina, siguiendo los derechos de los trabajadores, como prestaciones, seguro, cuota y demás. Ayuda en cuestiones prácticas a quienes lleva, a cabo dichos procedimientos, implementando cierta seguridad para proteger la integridad de la información.

**SCOPE**

Admin PI Es un sistema que funcionará como software para computo principalmente, que permitirá la administración, así como la consulta de los empleados referente a sus pagos.

Deberá contar con las actualizaciones de los estándares que manejan el SAT y el seguro, así como una vista general sobre los gastos de todo el negocio ante el personal.

**OBJECTIVES**

## Assumptions and Constraints

Las suposiciones y restricciones respecto del sistema, y que se derivan directamente de los requerimientos de la empresa son:

1. Uso de la metodología RUP
2. Uso de tecnología Microsoft
3. Uso de Netbeans
4. GUI
5. JUNIT
6. Wamp Server

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

## Project Deliverables

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

1. Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y **Plan de Desarrollo del Software**

Es el presente documento.

**Disciplina modelado del negocio**

1. **Modelo de Casos de Uso del Negocio**

Este modelo permite visualizar el alcance de la organización, representando lo que abarca y cuáles son sus límites. Así mismo, modela las actividades y procesos que ejecuta una organización, señala gráficamente las funciones y metas que persigue el negocio, y también permite identificar cuáles son los roles y entregables de la organización.

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.), permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representaremos con un diagrama de clases conceptuales.

**Disciplina ingeniería de requerimientos**

1. **Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

1. **Documento de especificación de requerimientos**

El objetivo de este artefacto es documentar todos los requerimientos del sistema, describir las funciones del sistema, los requerimientos no funcionales, las características del diseño y otros elementos necesarios para proporcionar una descripción completa y comprensiva de los requerimientos para el software a desarrollar.

**Disciplina análisis y diseño**

1. **Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Este modelo se basa en la descripción de elementos o usuarios externos al sistema (actores) y de la funcionalidad del sistema (casos de uso). Un modelo de casos de uso describe los requerimientos funcionales de un actor (usuarios, sistema, dispositivo, etc.) en términos que éste interactúa con el sistema. El modelo de caso de uso es una técnica efectiva y la ves simple para modelar los requerimientos del sistema desde la perspectiva del usuario. Este modelo lo representaremos con el modelo de casos de uso.

1. **Especificaciones de Casos de Uso**

Para efectos del presente proyecto, solamente se describirá el flujo de eventos principal para cada caso de uso. Se realizará una descripción breve y sencilla utilizando una plantilla de documento.

1. **Modelo de diseño**

Es una abstracción del Modelo de Implementación y su código fuente, el cual fundamentalmente se emplea para representar y documentar su diseño. Es usado como entrada esencial en las actividades relacionadas a implementación. Representa a los casos de uso en el dominio de la solución. El modelo de diseño lo representaremos con los diagramas de clases, diagramas de secuencia y el modelo de datos.

1. **Modelo de datos**

Describe la representación física y lógica de los datos constantes utilizados por la aplicación. Se utilizará siempre que se necesiten manejar datos constantes. Usualmente describirá los diferentes elementos componentes de la estructura de una base de datos relacional.

**Disciplina implementación**

1. **Modelo de implementación**

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes podemos encontrar datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. Para representar los diagramas del Modelo de Implementación emplearemos el diagrama de UML de Componentes.

**Disciplina pruebas**

1. **Plan de prueba**

Es la colección formada por los casos de prueba y procedimientos de prueba. Este artefacto incluye el propósito de las pruebas, qué elemento se va a probar, las herramientas a utilizar y con qué recursos, así como el documento que va hacer entregado. Al tener el resultado de las pruebas se puede comparar lo obtenido con lo esperado.

1. **Casos de prueba**

Este artefacto define un conjunto de datos de entradas, condiciones de ejecución y resultados esperados de las pruebas, identificados para hacer una evaluación de los aspectos específicos de un elemento objeto de prueba. Cada Caso de de Prueba está asociado a un escenario de un Caso de Uso en particular.

**Disciplina implantación**

1. **Sistema**

Este artefacto es el producto final, es decir, el sistema ya funcionando que puede ser instalado y ser utilizado por el cliente. Un Sistema se diferencia de una unidad de implantación, ya que el sistema puede contener varias unidades de implantación. Cabe destacar que dichas unidades de implantación que reúne el sistema pueden ser exportadas a una unidad de almacenamiento.

1. **Plan de implantación**

El objetivo principal de este artefacto es asegurar que el sistema llegue satisfactoriamente al conjunto de usuarios para el cual fue destinado. Este artefacto debe definir un conjunto de tareas que defina una transición sencilla para el cliente, para ello se debe minimizar el impacto que la implantación del sistema pueda llegar a causar en el personal del cliente, los sistemas de producción existentes y en todas las rutinas del negocio.

1. **Documentación para el usuario**
2. Este artefacto provee una ayuda a las personas que manipularán directamente el producto, acerca del uso que le debe dar al sistema y su instalación. Dicho artefacto debe ser discutido y aprobado por el cliente.

## Evolution of the Software Development Plan

*El plan de Desarrollo el Software se revisará semanalmente el día sábado, se reunirá el equipo y se revisará antes del comienzo de cada iteración en base al siguiente proceso:*

* + - 1. *Se revisa cada punto del plan de desarrollo.*
      2. *Se revisa que cada documentación se esté cumpliendo a cabalidad.*
      3. *Se anota en el cuadro de observaciones del plan de desarrollo.*
      4. *Antes de comenzar una iteración se toma en cuenta cada una de las observaciones realizadas.*
      5. *Se implementan las observaciones revisadas y rectificadas con el equipo al documento de plan de desarrollo*

|  |  |
| --- | --- |
| *NOMBRE DEL PUNTO A TRATAR* | *OBSERVACIONES* |
| *<<Aquí se detalla el nombre del punto a tratar>>* | *<<Aquí se detalla la observación 1 a realizar>>*  *<<Aquí se detalla la observación 2 a realizar>>*  *<<Aquí se detalla la observación 3 a realizar>>* |

# Project Organization

## Organizational Structure

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Descripción** | **Responsabilidades** |
| Adan Oswaldo Piceno | Líder del proyecto | Establecer las condiciones de trabajo  Dirigir y asignar recursos  Coordinar las interacciones con los clientes y usuarios finales  Planificar las iteraciones  Definir la organización del proyecto |
| Adan Oswaldo Piceno  Marlene Tapia | Analista de sistemas | Se encarga de dirigir el proceso de captura de requerimientos, definir los actores y casos de uso y estructurar el modelo de casos de uso, estableciendo la forma en que funcionará el sistema y cuáles son las restricciones del mismo. |
| Adan Oswaldo Piceno  Marlene Tapia | Diseñador de sistemas | Se encarga de la definición de la arquitectura que guiará el desarrollo, y de la continua refinación de la misma en cada iteración; debe construir cualquier prototipo necesario para probar aspectos riesgosos desde el punto de vista técnico del proyecto; definirá los lineamientos generales del diseño y la implementación. |
| Adan Oswaldo Piceno  Marlene Tapia | Desarrollador | Responsable de la codificación de de los componentes en código fuente en algún lenguaje de programación durante cada iteración  Responsable de las clases que ha desarrollado debiendo documentarlas, actualizarlas ante los cambios y mantenerlas bajo el control de la configuración de las mismas mediante la herramienta utilizada |

## External Interfaces

No Aplica

## Roles and Responsibilities

Esta matriz es una propuesta de las responsabilidades solamente con respecto a los artefactos considerados para este ejercicio.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Artefactos** | **Roles** | | | | |
| Líder de Proyecto | Analista | Diseñador | Desarrollador | Tester |
| Modelo de casos de uso de negocio | A | R |  |  |  |
| Doc. Visión | A | R |  |  |  |
| Doc. Especificación de requerimientos | A | R | A |  |  |
| Modelo de casos de uso | A | P | R |  |  |
| Especificación de casos de uso | A | R | A |  |  |
| Modelo de diseño | A | P | R |  |  |
| Modelo de datos | A | P | R |  |  |
| Modelo de implementación | A | A | A | R |  |
| Plan de prueba | A |  |  |  | R |
| Casos de prueba | A |  |  |  | R |
| sistema | A |  |  | R |  |
| Plan de implantación | A |  |  | R |  |
| Documentación para el usuario | A |  |  |  | R |

***Clave:*** R= Responsable; A = Aprobación necesaria; P = Participación necesaria;

# Management Process

## Project Estimates

1. En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | **Disciplinas/Actividades** | **Artefacto** | **Inicio** | **Fin** | | **Modelado del Negocio**   * + - Describir el negocio actual     - Desarrollar el modelo del dominio | * + - Modelo de casos de uso de negocio | 19-01-2019 | 02-02-2019 | | **Requerimientos**   * + - Realizar entrevistas     - Analizar el problema     - Clasificar y priorizar requerimientos | * + - Documento de visión     - Especificación de requerimientos de software | 26-01-2019 | 16-02-2019 | | **Análisis y Diseño**   * + - Especificación de casos de uso     - Realización de casos de uso     - Realización de los diagramas de clases     - Realización de los diagramas de secuencia     - Realización del modelo de datos     - Realizar prototipos | * + - Modelo de casos de uso     - Especificaciones de casos de uso     - Modelo de diseño     - Modelo de datos | 09-02-2019 | 06-04-2019 | | **Implementación**   * + - Estructurar el modelo de implementación     - Planificar la integración     - Implementar componentes | * + - Modelo de implementación | 23-02-2019 | 18-05-2019 | | **Pruebas**   * + - Definir misión de pruebas     - Validar estabilidad de componentes | * + - Plan de pruebas     - Casos de pruebas | 13-04-2019 | 11-05-2019 | | **Implantación**   * + - Planificar la implantación     - Desarrollar el material de apoyo | * + - Sistema     - Plan de implantación     - Documentación para el usuario |  | 25-05-2019 | |

## Project Plan

### Phase Plan

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Elaboración, Construcción y Transición solo fueron tomadas como referencias).

|  |  |  |
| --- | --- | --- |
| **Fase** | **No. Iteraciones** | **Duración** |
| Fase de Inicio | 1 | 3 semanas |
| Fase de Elaboración | 1 | 9 semanas |
| Fase de Construcción | 2 | 12 semanas |
| Fase de Transición | 2 | 4 semanas |

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

|  |  |  |  |
| --- | --- | --- | --- |
| **Descripción** | | **Hito** | |
| Fase de Inicio | | En esta fase desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente / usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase. | |
| Fase de Elaboración | | En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. | |
| Fase de Construcción | | Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 3.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta. | |
| Fase de Transición | | En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto. | |

### Iteration Objectives

*[List the objectives to be accomplished for each of the iterations.]*

### Releases

*[A brief description of each software release and whether it’s demo, beta, and so on.]*

### Project Schedule

*[Diagrams or tables showing target dates for completion of iterations and phases, release points, demos, and other milestones.]*

### Project Resourcing

#### Staffing Plan

*[Identify the numbers and type of staff required here, including any special skills or experience, scheduled by project phase or iteration.]*

#### Resource Acquisition Plan

*[Describe how you will approach finding and acquiring the staff needed for the project.]*

#### Training Plan

*[List any special training project team members will require, with target dates for when this training should be completed.]*

### Budget

*[Allocation of costs against the WBS and the Phase Plan.]*

## Iteration Plans

*[Each iteration plan will be enclosed in this section by reference.]*

## Project Monitoring and Control

### Requirements Management Plan

*[Enclosed by reference.]*

### Schedule Control Plan

*[Describe the approach taken to monitor progress against the planned schedule and how to take corrective action when required.]*

### Budget Control Plan

*[Describe the approach to be taken to monitor spending against the project budget and how to take corrective action when required.]*

### Quality Control Plan

*[Describe the timing and methods to be used to control the quality of the project deliverables and how to take corrective action when required.]*

### Reporting Plan

*[Describe internal and external reports to be generated, and the frequency and distribution of publication.]*

### Measurement Plan

*[Enclosed by reference.]*

## Risk Management Plan

*[Enclosed by reference.]*

## Close-out Plan

*[Describe the activities for the orderly completion of the project, including staff reassignment, archiving of project materials, post-mortem debriefings and reports, and so forth.]*

# Technical Process Plans

## Development Case

*[Enclosed by reference.]*

## Methods, Tools, and Techniques

*[List the documented project technical standards, etc., by reference:*

* *Business Modeling Guidelines*
* *User Interfaces Guidelines*
* *Use-Case-Modeling Guidelines*
* *Design Guidelines*
* *Programming Guidelines*
* *Test Guidelines*
* *Manual Style guide]*

## Infrastructure Plan

*[Enclosed by reference]*

## Product Acceptance Plan

*[Enclosed by reference]*

# Supporting Process Plans

## Configuration Management Plan

*[Enclosed by reference]*

## Evaluation Plan

*[As part of the* ***Software Development Plan,*** *this describes the project’s plans for product evaluation, and covers the techniques, criteria, metrics, and procedures used for evaluation— this will include walkthroughs, inspections, and reviews. Note that this is in addition to the Test Plan, which is not enclosed in the* ***Software Development Plan****.]*

## Documentation Plan

*[Enclosed by reference.]*

## Quality Assurance Plan

*[Enclosed by reference.]*

## Problem Resolution Plan

*[Enclosed by reference.]*

## Subcontractor Management Plan

*[Enclosed by reference.]*

## Process Improvement Plan

*[Enclosed by reference.]*

# Additional Plans

*[Additional plans if required by contract or regulations.]*

# Annexes

*[Additional material of use to the reader of the* ***Software Development Plan****.]*

# Index