



INSTITUTO TECNOLÓGICO  
“CORDILLERA”

CARRERA ANÁLISIS DE SISTEMAS

CONTROL DE PROCESOS DE INVENTARIO Y NÓMINA MEDIANTE UN  
SISTEMA WEB PARA LA EMPRESA CONSTRUCTUM S. A UBICADA EN LA  
CIUDAD DE QUITO

Proyecto de investigación previo a la obtención del título de tecnólogo en análisis de  
sistemas.

Autor: Peñafiel Asimbaya Bryan Patricio

Tutor: Ing. Hugo Heredia

Quito, Noviembre 2014

Declaración de aprobación de tutor y lector

## DECLARATORIA

Declaro que la investigación es absolutamente original, autentica, personal, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes. Las ideas, doctrinas resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

---

Peñafiel Asimbaya Bryan Patricio

1715761647

## CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante **BRYAN PATRICIO PEÑAFIEL ASIMBAYA**, por sus propios y personales derechos, a quien en lo posterior se le denominará el "CEDENTE"; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el "CESIONARIO". Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

**PRIMERA: ANTECEDENTE.-** a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado

**"ORGANIZACIÓN DE LA INFORMACIÓN EXTRAMURAL DE LOS GRUPOS SOCIALES, GENERADA POR LOS MÉDICOS DEL DISTRITO DE SALUD N17-D04, A TRAVÉS DE UNA APLICACIÓN WEB.**

**MÓDULO: CONTROL DE LOS ESTÁNDARES DE CALIDAD POR CICLOS DE VIDA DE LA INFORMACIÓN EXTRAMURAL"**, el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es

producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

**SEGUNDA: CESIÓN Y TRANSFERENCIA.-** Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita todos los derechos patrimoniales del programa de ordenador descrito en la cláusula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.).

El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

**TERCERA: OBLIGACIÓN DEL CEDENTE.-** El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

**CUARTA: CUANTIA.-** La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

**QUINTA: PLAZO.-** La vigencia del presente contrato es indefinida.

**SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.-** Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se solicitará la asistencia de un Mediador del Centro de Arbitraje y

Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de

Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas:

a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvenición, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

**SÉPTIMA: ACEPTACIÓN.-** Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 23 días del mes de Octubre del dos mil catorce.

f) \_\_\_\_\_  
C.C. N° 1715761647  
**Bryan Peñafiel**

f) \_\_\_\_\_  
Instituto Superior Tecnológico Cordillera  
**CESIONARIO**

## AGRADECIMIENTO

Primero agradezco a Dios por brindarme la vida, a mis padres por darme la oportunidad de realizarme como un profesional, a toda mi familia que siempre estuvo presente en cada victoria y derrota en especial a mi primo Mauricio.

A mi tutor Ing. Hugo Heredia y mi lectora Lic. Patricia Garzón que con sus conocimientos me apoyaron y me guiaron para culminar con éxito el presente proyecto.

## **DEDICATORIA**

A Dios por colmarme siempre de bendiciones y sabiduría.

A mis padres por brindarme todo su apoyo en los momentos más difíciles de mi carrera e instruir dignos ejemplos de superación y entrega

Bryan Patricio Peñafiel Asimbaya

## ÍNDICE GENERAL

Título	Página
Declaración de aprobación de tutor y lector .....	ii
Declaratoria .....	iii
Contrato de cesión sobre derechos propiedad intelectual .....	iv
Agradecimiento .....	viii
Dedicatoria .....	ix
Índice general .....	x
Índice de tablas .....	xiv
Índice de figuras .....	xvi
Resumen ejecutivo .....	xviii
Abstract .....	xx
Capítulo I: Antecedentes .....	1
1.01. Contexto .....	1
1.02. Justificación .....	2
1.03. Definición del Problema Central .....	2
Capítulo II: Análisis de involucrados .....	7
2.01. Requerimientos .....	7
2.01.01. Descripción del sistema actual .....	7
2.01.02. Visión y Alcance .....	7
2.01.03. Entrevistas .....	9



2.01.04. Matriz de requerimientos .....	10
2.01.05. Descripción Detallada de Requerimientos .....	10
2.02. Mapeo de Involucrados .....	16
2.03. Matriz de Involucrados .....	16
Capítulo III: Árbol de problemas-objetivos .....	18
3.01. Árbol de Problemas .....	18
3.02. Árbol de objetivos .....	19
3.03. Diagramas de casos de uso .....	20
3.04. Casos de uso de realización .....	20
3.05. Diagrama de secuencias del sistema .....	28
3.06. Especificación de Casos de Uso .....	31
Capitulo IV: Análisis de alternativas .....	34
4.01. Matriz de Análisis de alternativas .....	34
4.02. Matriz de Impactos de Objetivos .....	35
4.03. Estándares para el Diseño de Clases .....	36
4.04. Diagrama de Clases .....	37
4.05. Modelo Lógico – Físico .....	37
4.06. Diagrama de Componentes .....	37
4.07. Diagrama de Estrategias .....	38
4.08. Matriz de Marco Lógico .....	39
4.09. Vistas Arquitectónicas .....	40



4.09.01. Vista lógica.....	40
4.09.02. Vista física.....	41
4.09.03. Vista de desarrollo .....	42
4.09.04. Vista de procesos .....	42
Capítulo V: Propuesta .....	44
5.01. Especificación de estándares de programación.....	44
5.02. Diseño de Interfaces de Usuario .....	45
5.03. Especificación de pruebas de unidad .....	48
5.04. Especificación de pruebas de aceptación .....	49
5.05. Especificación de pruebas de carga.....	50
5.06. Configuración del Ambiente mínima/ideal.....	50
Capítulo VI: Aspectos administrativos .....	52
6.01. Recursos .....	52
6.02. Presupuesto .....	53
6.03. Cronograma.....	53
Capítulo VII: Conclusiones y recomendaciones .....	54
7.01. Conclusiones .....	54
7.02. Recomendaciones.....	55
Anexos .....	56
A.0 1 Descripción detallada de la matriz de requerimientos .....	57
A.0 2 Diagrama de clases.....	59



A.0 3 Modelo Físico-Lógico .....	60
A.0 4 Parámetros de la programación. ....	62
A.0 5 Script de la base de datos .....	64
Manual de usuario .....	98
Manual técnico .....	115
Manual de instalación .....	144
Bibliografía .....	150

## ÍNDICE DE TABLAS

Tabla 1. Matriz de Fuerzas T .....	3
Tabla 2. Entrevista dirigida a Gerencia.....	9
Tabla 3. Descripción de requerimientos RF001.....	10
Tabla 4. Descripción de requerimientos RF002.....	11
Tabla 5. Descripción de requerimientos RF003.....	11
Tabla 6. Descripción de requerimientos RF004.....	12
Tabla 7. Descripción de requerimientos RF005.....	12
Tabla 8. Descripción de requerimientos RF006.....	13
Tabla 9. Descripción de requerimientos RF007.....	13
Tabla 10. Descripción de requerimientos RNF001.....	14
Tabla 11. Descripción de requerimientos RNF002.....	14
Tabla 12. Descripción de requerimientos RNF003.....	15
Tabla 13. Matriz de involucrados Directos e Indirectos. ....	17
Tabla 14. Especificación de diagrama de realización UCR001.....	21
Tabla 15. Especificación de diagrama de realización UCR002.....	22
Tabla 16. Especificación de diagrama de realización registro del personal.....	23
Tabla 17. Especificación de diagrama de realización UCR004.....	24
Tabla 18. Especificación de diagrama de realización UCR005.....	25
Tabla 19. Especificación de diagrama de realización UCR006.....	26
Tabla 20. Especificación de diagrama de realización UCR007.....	27
Tabla 21. Especificación de caso de uso UC001. ....	31



Tabla 22. Especificación de caso de uso UC002. ....	32
Tabla 23. Especificación de caso de uso UC003 .....	32
Tabla 24. Especificación de caso de uso UC004 .....	32
Tabla 25. Especificación de caso de uso UC005 .....	33
Tabla 26. Especificación de caso de uso UC006 .....	33
Tabla 27. Especificación de caso de uso UC007 .....	33
Tabla 28. Matriz de análisis de alternativas. ....	34
Tabla 29. Análisis del impacto de objetivos. ....	35
Tabla 30. Parámetros para el diseño del Diagrama de clases .....	36
Tabla 31. Matriz marco lógico. ....	39
Tabla 32. Controles. ....	44
Tabla 33. Especificación de la prueba de unidad PUD001. ....	48
Tabla 34. Especificación de la prueba de unidad PUD002. ....	48
Tabla 35. Especificación de la prueba de aceptación EPA001. ....	49
Tabla 36. Especificación de la prueba de aceptación EPA002. ....	49
Tabla 37. Especificación de la prueba de carga PDCD001. ....	50
Tabla 38. Recurso Humano. ....	52
Tabla 39. Recurso Material. ....	52
Tabla 40. Presupuesto. ....	53

## ÍNDICE DE FIGURAS

Figura 1. Mapeo de involucrados.....	16
Figura 2: Árbol de problemas .....	18
Figura 3: En el árbol de objetivos .....	19
Figura 4: Diagrama de casos de uso general.....	20
Figura 5: .Diagrama de realización control de inventarios .....	21
Figura 6: Diagrama de realización manejo de nómina.....	22
Figura 7: Diagrama de realización registro del personal .....	23
Figura 8: Diagrama de realización mostrar reportes .....	24
Figura 9: Diagrama de realización control de acceso .....	25
Figura 10: Diagrama de realización registro de documentos.....	26
Figura 11: Diagrama de realización código Qr.....	27
Figura 12: Diagrama de secuencia de control de inventario .....	28
Figura 13: .Diagrama de secuencia manejo de nómina.....	28
Figura 14: Diagrama de secuencia registro de personal.....	29
Figura 15: Diagrama de secuencia mostrar reporte.....	29
Figura 16: Diagrama de secuencia control de acceso .....	30
Figura 17: Diagrama de secuencia registro de documentos.....	30
Figura 18: Diagrama de secuencia Códigos Qr.....	31
Figura 19: Diagrama de componentes del sistema.....	37
Figura 20: Detalle del diagrama de estrategias .....	38
Figura 21: Detalle de la Vista lógica del sistema.....	40



Figura 22: Detalle de la Vista física del sistema.....	41
Figura 23: Detalle de la Vista de desarrollo.....	42
Figura 24: Detalle de actividades de bodega .....	43
Figura 25: Detalle del diagrama de nómina .....	43
Figura 26: Interfaz del sistema.....	45
Figura 27: Interfaz del personal. ....	46
Figura 28: Interfaz del personal. ....	47
Figura 29: Cronograma. ....	53

## RESUMEN EJECUTIVO

La finalidad de diseñar una aplicación informática que está orientada a la web se basa en automatizar el control de procesos de inventarios y nómina de la empresa "CONSTRUCTUM S.A".

El capítulo I consiste en describir el origen del proyecto, la situación de la empresa, los procesos actuales de la empresa, los factores que intervinieron para impulsar este proyecto. También permite evaluar las diversas situaciones que motivan y bloquean en el desarrollo del software.

En el capítulo II se delimita el alcance del proyecto, detallando los puntos que se van a cubrir. Es importante mencionar que en esta parte del proyecto se realiza un análisis que permite determinar a las personas favorecidas de manera directa o indirecta, al aplicar el sistema. Estas personas son entrevistadas por la persona responsable de desarrollar el proyecto, las preguntas entrevistadas deben quedar documentadas en esta parte del proyecto físico.

El capítulo III permite analizar de una manera mucho más detallada las falencias del sistema actual y plantearnos nuevos objetivos, tratando de disminuir la ineficacia de los procesos actuales. Se documentara también los casos de uso, los cuales son una forma gráfica de demostrar la participación de cada uno de los objetos en el sistema, así como también ayudarán a identificar los procesos que realizara cada uno de los usuarios y sus delimitaciones.

En el capítulo IV se plantearán soluciones alternativas que pueden llegar a ser estrategias para el proyecto, así como también mediante la matriz de marco lógico detallar los resultados que se espera del proyecto. Y el modelado UML.

El capítulo V está basado en demostrar los parámetros de programación, y todo lo referente al desarrollo del software como por ejemplo diseño de interfaces, base de datos, pruebas de seguridad, pruebas de carga, simulaciones, entre otras evaluaciones, las cuales hacen q nuestra aplicación sea confiable.

En el capítulo VI se detallan los recursos que utilizamos para realizar el proyecto, costos, el presupuesto y queda expresado el cronograma de actividades que expresa el tiempo que se empleó para desarrollar el software y su respectiva documentación.

El capítulo VII lleva las recomendaciones, conclusiones. Estos parámetros sirven para dejar expresada de manera escrita instrucciones para el correcto uso de la aplicación.

## ABSTRACT

The purpose of designing a software application that is focused on the web is based on automating process control inventory and payroll company "CONSTRUCTUM SA".

Chapter I is to describe the origin of the project, the status of the company, existing business processes, factors intervened to promote this project. It also allows to evaluate the different situations that motivate and block in the development of software.

In Chapter II the project scope is delimited, detailing the points to be covered. It is noteworthy that in this part of the project an analysis to determine the advantaged direct or indirect way to implement the system is performed. These people are interviewed by the person responsible for developing the project, the interviewees questions should be documented in this part of the physical project.

Chapter III to analyze in a more detailed manner the shortcomings of the current system and set new goals, trying to reduce the inefficiency of the current process.

Use cases, which are graphically, demonstrate the participation of each of the objects in the system, as well as help identify the processes that perform each of the users and their boundaries are well documented.

In Chapter IV workarounds that can become strategies for the project will be raised, as well as through the logical framework matrix detailing the results expected from the project. And UML modeling.

Chapter V is based on demonstrating the programming parameters, and everything related to software development such as interface design, database, security testing, load testing, simulations, among other measures, which make our application is reliable.



In Chapter VI we use resources to make the project cost, budget and expressed is the schedule of activities expressing the time it took to develop the software and related documentation are described.

Chapter VII takes the recommendations and conclusions. These parameters are expressed to leave written instructions for proper use of the application.

## Capítulo I: Antecedentes

### 1.01. Contexto

El control de procesos de inventarios y nómina es uno de los procesos más importantes en los diversos tipos de empresa, siendo manejada de esta manera en la mayoría de compañías. El área de recursos humanos es el encargado de evaluar y contratar al personal, por lo que es necesario para llevar la nómina detallada de cada uno de los empleados, como nombres, títulos, salario, cargo, etc. El bodeguero es el encargado de llevar el inventario de los materiales. Estos registros en la actualidad son llevados en formularios realizados en hojas de cálculo (Excel), en donde se detalla información de recepción y entrega de materiales, por lo cual esto ha sido una debilidad al momento de obtener información acerca del stock de los materiales. Bajando su nivel de eficacia al momento del despacho de materiales.

Con el transcurso del tiempo se ha dado grandes giros tecnológicos en todo ámbito, dando así a las empresas cambio e innovación, ofreciendo atención de calidad, y mejora en sus servicios, brindando a las diferentes empresas de manera interna como externa una imagen renovada, aumentando su actividad productiva, organización y producción.

Con estos parámetros la intención al sistematizar la forma de llevar los diferentes registros es mejorar el proceso de control del inventario de bodega, dando un rápido acceso a la información mediante reportes, así como mejorar la organización en el departamento de recursos humanos garantizando la integridad de la información de los empleados como de la empresa.

## **1.02. Justificación**

En busca de automatizar los procesos de inventario y nómina se propone la creación de un sistema que automatice el control de las mismas, el cual será diseñado específicamente para personas que tengan conocimiento de informática básica. Con esto lograremos que los usuarios obtengan reportes de una forma sencilla y eficaz. Evitando posibles fallas en el sistema ocasionadas por el manejo de usuario.

El proyecto se lo va a realizar pensando en los beneficios que va a adquirir la empresa al ir complementando sus estrategias de producción, y empleando nuevas técnicas para el registro y clasificación de datos.

Considerando que la empresa obtendrá seguridad en cuanto a la información de registros financieros como el sueldo de los empleados y su información personal, así mismo como del registro de materiales y el stock que tiene cada uno de estos, aplicando seguridades en el rol de usuario.

A esto añadiendo otro tipo de beneficios como una mejor recepción y clasificación de materiales los cuales vienen de distintos proveedores, hacer un reporte con el tiempo considerable para realizar una compra oportuna de materiales, permitiendo analizar nuevos proveedores. Cabe mencionar mejorar la atención del cliente, y un pago puntual de los salarios a todos los empleados. La intención a aplicar el sistema es proporcionar información actualizada y detallada en menor tiempo que el que se emplea actualmente para acceder al mismo tipo de información.

## **1.03. Definición del Problema Central**

La matriz T nos ayuda a saber la situación actual de la empresa y da a conocer las falencias y necesidades que suceden dentro de la misma con el fin de cooperar con una aplicación informática.

Tabla 1

*Matriz de Fuerzas T*

ANALISIS DE FUERZAS T					
Situación Empeorada	Situación Actual				Situación Mejorada
Alto índice de pérdidas de inventario y equivocación de generación de nómina.	Inadecuado control de nómina e inventario en la empresa CONSTRUCTUM.				Control adecuado de nómina e inventario en la empresa CONSTRUCTUM.
Fuerzas Impulsadoras	<b>I</b>	<b>PC</b>	<b>I</b>	<b>PC</b>	<b>Fuerzas Bloqueadoras</b>
Implementación de tecnología	3	5	2	3	Falta de recursos financieros para la aplicación del sistema.
Registro correcto del inventario.	3	5	3	3	Desorganización del inventario de bodega.
Establecer generaciones exactas de nómina.	3	4	3	3	Deficiencia en la creación de nómina del empleado.
Facilitar la búsqueda de materiales y herramientas del inventario mediante códigos a	4	4	4	5	Falta de conocimientos del proceso de escaneo de códigos Qr.
Establecer un informe detallado sobre el inventario de bodega.	3	5	3	3	Inadecuado registro de informes.
Registro de la documentación referente a los pagos de bodega.	5	5	2	2	Deficiencia en el manejo de documentación.

*Nota*

I==Intensidad es el nivel de impacto de la fuerza sobre la problemática actual.

PC = Potencial de Cambio logra modificar o aprovechar la fuerza para llegar a la situación deseada

Rangos de medición de la Matriz de Fuerzas T.

1 = Bajo

2 = Medio Bajo

3 = Medio

4 = Medio Alto

5 = Alto

En la matriz de fuerza se especifica las situaciones: empeorada, actual y mejorada mediante la cual detallaremos los extremos a evadir, la problemática en el presente y lo que se desea hacer en nuestra investigación.

### **Fuerzas impulsadoras:**

#### **Implementación de tecnología.**

I=3 Medio, Equipo tecnológico en buen estado y respaldo económico de la empresa para costear los materiales que necesitaremos al momento de la aplicación del software.

PC=5 alto, Permitirá mejorar la organización tanto de los inventarios como de la nómina de empleados de cada uno de los departamentos involucrados favorecidos en la automatización.

#### **Registro correcto del inventario.**

I=3 medio, Actualmente la empresa no cuenta con un mecanismo de registro factible para llevar un control de inventario y evitar pérdidas.

PC=5 Alto, al aplicar el sistema nos permitirá actualizar los materiales y herramientas del inventario dando resultados inmediatos y precisos en un tiempo adecuado.

#### **Generar nominas confiables.**

I=3 Medio, Al momento de revisar la información de los empleados pudimos observar algunos inconvenientes como información errónea, redundante, entre otros.

PC=4 Medio Alto, al registrar en el sistema la información de cada uno de los empleados se busca una mejor organización, tomando en cuenta la seguridad de los registros.

#### **Facilitar la búsqueda de materiales y herramientas del inventario mediante códigos a Qr.**

I=3 Medio, por el momento en mecanismo de búsqueda de materiales se lo hace de manera manual, en donde el bodeguero es el encargado de registrar los materiales de bodega tanto las entradas como salidas .

PC=4 Medio Alto, al automatizar el proceso de búsqueda se pretende realizar un mejor control del inventario ya que mediante el escanero del código Qr mediante una cámara web nos mostrara toda la información del material o herramienta.

**Establecer un informe detallado sobre el inventario de bodega.**

I=4 Medio Alto, para obtener un informe detallado de la bodega se debe recopilar toda la información que es registrada en hojas lo cual hace que este proceso se prolongue.

PC=5 Alto, el sistema va a generar un reporte inmediato del inventario de bodega tomando la información registrada en la base de datos.

**Registro de la documentación referente a los pagos de bodega.**

I=4 Medio Alto, el registro de este tipo de documentos es muy importante para la empresa ya que esta involucrada la parte financiera de la empresa.

PC=4 Medio Alto, Al automatizar este registro se logra organizar la información y ayudar a respaldar los registros de contabilidad.

**Fuerzas bloqueadoras**

**Falta de recursos financieros para la aplicación del sistema.**

I=2 Medio Bajo, el crecimiento de la empresa ha ocasionado gastos por lo que no se ha implementado un sistema de automatización.

PC=3 Medio, al implementar un sistema obtendrían beneficios notables como mejorar el registro y acceso a la información.

**Desorganización del inventario de bodega.**

I=3 Medio, se han encontrado problemas como pérdida de registros, documentación desorganizada, falta de materiales.

PC=3 Medio Bajo, dificultad al obtener información exacta de bodega.

### **Deficiencia en la creación de nómina del empleado.**

I=3 Medio, la redundancia de información, registros erróneos han implicado que la creación de nóminas no sea confiable.

PC=3 Medio, pocos recursos con los que cuenta la institución para el desarrollo de la aplicación informática.

### **Falta de conocimientos del proceso de escaneo de códigos Qr.**

I=4 Medio Alto, falta de conocimiento sobre el sistema de escaneo Qr.

PC=5 Alto, capacitación sobre el uso del sistema a los empleados que intervienen en el manejo del mismo.

### **Inadecuado registro de informes.**

I=3 Medio, Mal manejo de la herramienta ofimática EXCEL.

PC=3 Medio, información inexacta al manejar mal la herramienta ofimática.

### **Deficiencia en el manejo de documentación.**

I=2 Medio Bajo, descuido de documentos físicos que producen pérdidas de los mismos.

PC=2 Medio Bajo, reportes incompletos por falta de un control de documentación.

## Capítulo II: Análisis de involucrados

### 2.01. Requerimientos

#### 2.01.01. Descripción del sistema actual

La Empresa "Constructum S.A" actualmente lleva los informes de bodega en hojas de Excel. Esta información es manejada por el jefe de bodega en donde se registran los materiales de construcción, herramientas e información detallada de los materiales los cuales son ubicados en su respectivo espacio. Este método de llevar la información no cumple con la seguridad que debería tener ya que puede ocasionar muchos problemas al momento de generar un reporte al supervisor o realizar un control de los materiales, sea por información redundante, pérdida de información, mala digitación de la información, entre otros.

La nómina del personal se maneja en hojas de cálculo (EXCEL), este reporte lleva los nombres y la información detallada de la remuneración de cada empleado. Dicha información es entregada a la persona encargada de la contabilidad de la empresa. El procedimiento de cálculo de salarios que llevan no es confiable ya que general mal la nómina, ya que existen muchos descuidos de las mismas y esto conlleva a pérdidas financieras.

#### 2.01.02. Visión y Alcance

##### Visión

El sistema de procesos de inventarios y nómina mediante un sistema web permitirá obtener una herramienta útil para el registro de datos, permitiendo manejar los registros de manera oportuna, y con la seguridad que debe poseer un sistema informático.

## Alcance

El resultado que se desea obtener al desarrollar la aplicación es generar informes de bodega y nómina de empleados. Permitiendo un mejor desempeño en los labores del bodeguero y del personal de contabilidad que maneja nómina. Asegurando la integridad de información ingresada en la base de datos.

El sistema tendrá los siguientes módulos:

- Módulo de seguridad

El sistema otorgará permisos a cada usuario según el cargo y su rol, el ingreso del usuario al sistema será a través de un login que contendrá un ingreso de usuario y password entregados a cada empleado desde administración.

- Módulo de Mantenimiento

El sistema permitirá actualizar la información de inventarios y realizar cálculos de la nómina del empleado.

- Módulo de Procesos

Las técnicas para controlar el material y herramientas serán a través de códigos Qr con el que se les podrán identificar el detalle de cada uno ellos, a través de un scanner mediante una webcam, además de una pantalla visual la cual se la podrá realizar manualmente todo el proceso anterior.

Los usuarios encargados de la nómina de personal podrán registrar los pagos de los sueldos del empleado, bonificaciones, contratos, permisos, multas, y deducciones.

- Módulo de Reportes y Consultas

El usuario tendrá opción de búsqueda y como resultado obtendrá un registro de la petición dada por el usuario.

### 2.01.03. Entrevistas

La siguiente entrevista brindará información sobre la funcionalidad del sistema actual, organización de la unidad, cargos y roles de los empleados con el fin de centrar las bases para el desarrollo del sistema.

Tabla 2

#### *Entrevista dirigida a Gerencia*

Identificador: 001		
Preguntas	Objetivos	Análisis Posterior
¿Qué procesos desea controlar en el sistema?	Determinar los procesos que se van a controlar	<ol style="list-style-type: none"> <li>1. Controlar el manejo de inventario de bodega.</li> <li>2. Manejar la nómina del empleado.</li> <li>3. En el sistema se deberá ingresar la información de los empleados.</li> <li>4. El sistema deberá mostrar reportes del inventario.</li> <li>5. El sistema debe tener un login y un password para cada tipo de usuario.</li> <li>6. El sistema deberá permitir registrar la documentación referente a los pagos de bodega.</li> </ol>
¿Cuáles usuarios intervienen en el sistema?	Conocer los cargos de cada usuario que intervendrá en el sistema	<ol style="list-style-type: none"> <li>1. El administrador deberá tener acceso y control de todo el sistema</li> <li>2. El control de inventario será otorgado al bodeguero de la empresa</li> <li>3. El empleado de T.T.H.H será quien maneje la nómina del empleado en el sistema.</li> </ol>
¿Conoce los procesos de control con códigos QR?	Determinar el nivel de conocimientos sobre códigos QR	<ol style="list-style-type: none"> <li>1. Implementar la inserción de códigos QR a los materiales y herramientas de bodega.</li> </ol>

*Nota:* La información que se obtuvo mediante la entrevista serán para los requerimientos del sistema.

Durante el proceso de la entrevista se llega a la conclusión que la empresas sin importar su giro, tamaño o magnitud, no con lleva un sistema propio para el control y manejo de bodega y personal, por lo cual se debe tomar en cuenta el propósito de un control del inventario y nómina, para evitar la pérdida económica en material, herramienta, además de la información de los trabajadores del negocio.

#### 2.01.04. Matriz de requerimientos

La matriz de requerimientos da a conocer las necesidades de la empresa para poder automatizarlas en el sistema (Ver A.01).

#### 2.01.05. Descripción Detallada de Requerimientos

Tabla 3

Descripción de requerimientos RF001

El sistema debe permitir manejar el inventario	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RF001	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Stock de material y herramienta	
<b>Descripción</b>	El jefe de bodega podrá agregar, modificar y eliminar la información del inventario.	
<b>Datos de Salida</b>	Información actualizada de inventario.	
<b>Resultados Esperados</b>	Reporte	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador Bodeguero	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1.- Para ingresar al sistema el Bodeguero debe estar registrado en la base de datos	
<b>Poscondiciones</b>		
<b>Criterios de Aceptación</b>	1.-Delimitar los bloques de seguridad de cada uno de los roles de usuario.	

Tabla 4

*Descripción de requerimientos RF002*

El sistema debe poder generar la nómina del empleado	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RF002	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Datos referentes a pagos del empleado.	
<b>Descripción</b>	El jefe de T.T.H.H conseguirá generar la nómina de los empleados	
<b>Datos de Salida</b>	Nómina de los empleados.	
<b>Resultados Esperados</b>	Nómina integra.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador T.T.H.H	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1.- Para ingresar al sistema el Bodeguero debe estar registrado en la base de datos	
<b>Poscondiciones</b>		
<b>Criterios de Aceptación</b>	1.-Delimitar los bloques de seguridad de cada uno de los roles de usuario.	

Tabla 5

*Descripción de requerimientos RF003*

El sistema podrá registrar los datos del personal	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RF003	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Información del empleado.	
<b>Descripción</b>	El administrador será capaz de actualizar la lista de empleados y la información de cada uno.	
<b>Datos de Salida</b>	Información de empleados actualizada.	
<b>Resultados Esperados</b>	Reporte de cada empleado de la empresa.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1.- Para ingresar al sistema el administrador debe estar registrado en la base de datos	
<b>Poscondiciones</b>		
<b>Criterios de Aceptación</b>	1.-Delimitar los bloques de seguridad de cada uno de los roles de usuario	

Tabla 6

*Descripción de requerimientos RF004*

El sistema deberá mostrar reportes de cada material y herramienta del inventario	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RF004	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Código de material y herramienta deseado.	
<b>Descripción</b>	El administrador y bodeguero visualizaran el reporte actualizado de cada material y herramienta que lo requieran.	
<b>Datos de Salida</b>	Reporte seguro y confiable.	
<b>Resultados Esperados</b>	Dar a conocer sobre la situación actual de cada material y herramienta del inventario.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador Bodeguero	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1. El material-herramienta debe estar registrado.	
<b>Poscondiciones</b>	Ninguna	
<b>Criterios de Aceptación</b>	Al momento de requerir información de los materiales-herramientas, el sistema se encargara de emitir un reporte con la información asociada al mismo.	

Tabla 7

*Descripción de requerimientos RF005*

El sistema debe restringir el acceso según el rol y el cargo	<b>Estado</b>	<b>Análisis</b>
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RF005	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Login de usuario y password.	
<b>Descripción</b>	Al ingresar el login y password de usuario se habilitara los procesos respectivos según su función dentro del sistema.	
<b>Datos de Salida</b>	Limitaciones según el rol del usuario	
<b>Resultados Esperados</b>	Control de acceso en el sistema.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador Jefe T.T.H.H Jefe de bodega	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	Debe estar registrado cada uno de los usuarios con su respectivo rol en la base de datos	
<b>Poscondiciones</b>	Ninguna	
<b>Criterios de Aceptación</b>	Módulos de seguridad de cada uno de los usuarios con password encriptados.	

Tabla 8

*Descripción de requerimientos RF006*

El sistema deberá permitir registrar la documentación referente a los pagos de bodega.			
<b>Creador por</b>	Bryan Peñafiel	Actualizado por	Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización	09/07/2014
<b>Identificador</b>	RF006		
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento	Funcional
<b>Datos de Entrada</b>	Registro de facturas		
<b>Descripción</b>	El jefe de T.T.H.H ingresara los datos de las facturas de bodega		
<b>Datos de Salida</b>	Formulario para el ingreso de documentos		
<b>Resultados Esperados</b>	Información guardada en la base de datos		
<b>Origen</b>	Administrador		
<b>Dirigido a</b>	Administrador T.T.H.H Bodeguero		
<b>Prioridad</b>	Alta		
<b>Requerimientos Asociados</b>	Ninguno		
<b>ESPECIFICACIÓN ninguna</b>			
<b>Precondiciones</b>	Obtener la factura.		
<b>Poscondiciones</b>	Ninguna		
<b>Criterios de Aceptación</b>	Verificar si la información ha sido guardada satisfactoriamente.		

Tabla 9

*Descripción de requerimientos RF007.*

El sistema debe permitir la creación y escaneo de códigos Qr.		Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por	Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización	09/07/2014
<b>Identificador</b>	RF007		
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento	Funcional
<b>Datos de Entrada</b>	Código Qr del material o herramienta solicitado.		
<b>Descripción</b>	El administrador y bodeguero mediante una cámara web conseguirán escanear el código del material o herramienta y se les desplegara el reporte del mismo.		
<b>Datos de Salida</b>	Reporte de material o herramienta.		
<b>Resultados Esperados</b>	El administrador al momento de crear un nuevo material o herramienta se generará automáticamente su código Qr para una búsqueda más eficiente ahorrando tiempo al momento de requerir un reporte.		
<b>Origen</b>	Administrador		
<b>Dirigido a</b>	Administrador Bodeguero		
<b>Prioridad</b>	Alta		
<b>Requerimientos Asociados</b>	Ninguno		
<b>ESPECIFICACIÓN</b>			
<b>Precondiciones</b>	Roles registrados en el sistema		
<b>Poscondiciones</b>	Crear un código para distinguir los materiales		
<b>Criterios de Aceptación</b>	Resguardo de la información del sistema		

Tabla 10

*Descripción de requerimientos RNF001.*

El sistema debe ser de fácil navegabilidad.	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RNF001	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Ingreso del usuario a la aplicación	
<b>Descripción</b>	Sistema desarrollado para usuarios con conocimientos básicos sobre informática básica.	
<b>Datos de Salida</b>	Manejo de la aplicación	
<b>Resultados Esperados</b>	El sistema debe ser manipulado con facilidad apegado a las restricciones de usuario	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador T.T.H.H Bodeguero	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1. Ingresar al sistema 2. Seleccionar rol de usuario	
<b>Poscondiciones</b>	Ninguno	
<b>Criterios de Aceptación</b>	Tareas realizadas satisfactoriamente	

Tabla 11

*Descripción de requerimientos RNF002*

El sistema debe ser seguro y mantener la integridad de la información	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RNF002	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	1.- seleccionar rol de usuario	
<b>Descripción</b>	1.- El sistema debe tener password encriptados, lo cual impide el acceso a la información de usuarios ajenos.	
<b>Datos de Salida</b>	Los usuarios obtendrán distintos módulos de acceso y manipulación.	
<b>Resultados Esperados</b>	Formularios permitidos para los distintos usuarios.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador T.T.H.H Bodeguero	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	1. Información registrada	
<b>Poscondiciones</b>	Ninguna	
<b>Criterios de Aceptación</b>	Aplicar bloques de seguridad.	

Tabla 12  
Descripción de requerimientos RNF003

El sistema debe controlar el acceso a los diferentes módulos.	Estado	Análisis
<b>Creador por</b>	Bryan Peñafiel	Actualizado por Bryan Peñafiel
<b>Fecha de Creación</b>	07/07/2014	Fecha de Actualización 09/07/2014
<b>Identificador</b>	RNF003	
<b>Estado de Requerimiento</b>	Crítico	Tipo de Requerimiento Funcional
<b>Datos de Entrada</b>	Rol de usuario	
<b>Descripción</b>	1.- como son distintos tipos de usuarios los que ingresan a la aplicación se han tomado en cuenta bloques de acceso los cuales se encargan de otorgar los permisos de acceso a cada uno de los roles de usuario	
<b>Datos de Salida</b>	Formularios para cada tipo de usuario	
<b>Resultados Esperados</b>	Delimitar el acceso.	
<b>Origen</b>	Administrador	
<b>Dirigido a</b>	Administrador T.T.H.H Bodeguero	
<b>Prioridad</b>	Alta	
<b>Requerimientos Asociados</b>	Ninguno	
<b>ESPECIFICACIÓN</b>		
<b>Precondiciones</b>	Al momento de crear usuarios se elegirá el rol	
<b>Poscondiciones</b>	Ninguno	
<b>Criterios de Aceptación</b>	Restringe el acceso a la información de la empresa.	

## 2.02. Mapeo de Involucrados

El mapeo de los involucrados es un instrumento fundamental que permite identificar a aquellos funcionarios y autoridades que pueden contribuir u obstaculizar con el logro de los objetivos determinados de la aplicación en el desarrollo e implementación del proyecto.

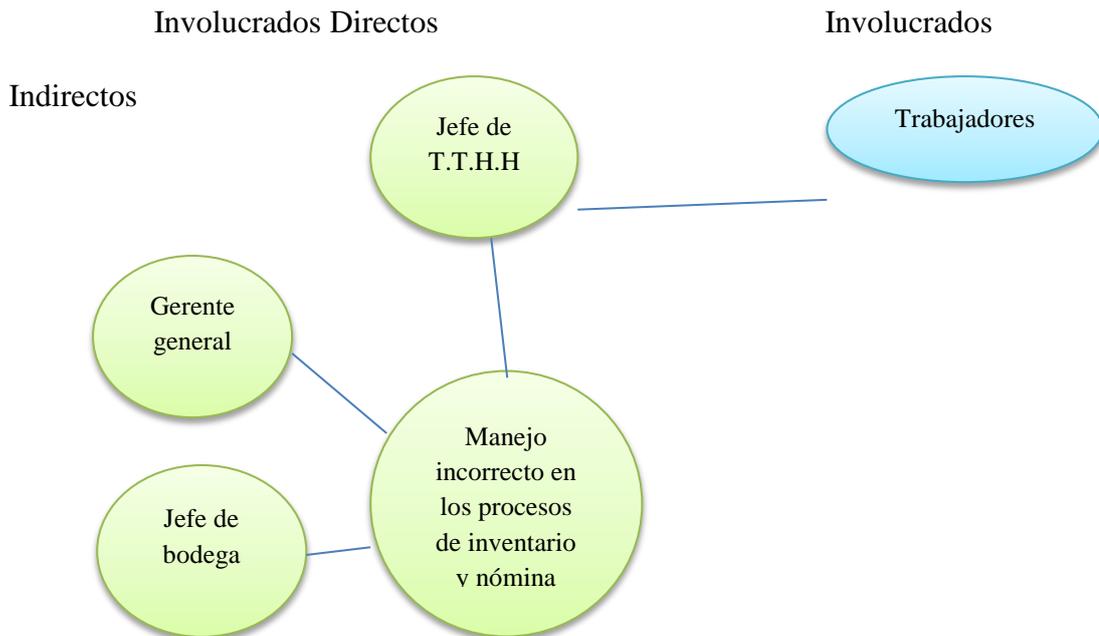


Figura 1. Mapeo de involucrados. Indica a los actores involucrados y no involucrados que intervienen en el objetivo del proyecto.

## 2.03. Matriz de Involucrados

La matriz de involucrados describirá detalladamente los actores involucrados dentro del problema a resolver como cuáles son sus intereses sobre el problema central esto ayudará a conocer cuál es la necesidad primordial dentro de la empresa, mediante los problemas percibidos sabremos las causas y razones por la cual se necesita de un sistema informático, recursos mandatos y capacidades a través de ello se conocerá las funciones de cada actor, intereses sobre el proyecto indicando en donde se involucrará cada actor en el sistema y en los conflictos potenciales obtendremos

información sobre los riesgos y daños que puede tener la empresa sino se realiza un control.

Tabla 13

*Matriz de involucrados Directos e Indirectos.*

Actores Involucrados	Interés sobre el problema central	Problemas percibidos	Recursos mandatos y capacidades	Intereses sobre el proyecto	Conflictos potenciales
Gerente general	Obtener un control óptimo sobre las áreas de T.T.H.H y Bodega	No cuenta con el control adecuado para la prevención de pérdidas económicas y de información.	Brindar herramientas tecnológicas a distintas áreas de la empresa	Implementación de la tecnología en la empresa.	La ausencia de un sistema informático.
T.T.H.H.	Generar una nómina confiable de los empleados que laboran en la empresa.	Nóminas generadas de forma inadecuada.	Contribución de documentación de todos los procesos con los trabajadores.	Implementación de un aplicativo informático que facilite la generación de nóminas.	Descuentos indebidos en su rol de pagos. Bonificaciones que no corresponden a los empleados.
Bodeguero	Tener un registro correcto del inventario de bodega	No tiene un inventario completo de los materiales y herramientas que existen en la empresa.	Contribución de documentación de todos los procesos con el material, herramienta	Control adecuado de los materiales y herramientas con los que cuenta la empresa.	Pérdidas en el inventario de bodega.
Trabajadores	Conseguir que los pagos sean los respectivos	Desconocimiento de las remuneraciones que se debe pagar a un empleado.	Cumplir con sus labores respectivas.	Confiable de sus pagos y remuneraciones	Perdida de remuneraciones

### Capítulo III: Árbol de problemas-objetivos

#### 3.01. Árbol de Problemas

El árbol de problemas es una manera de analizar la problemática se construye a partir el problema central, los cuales se van analizando y enlazándolos en relación a la causa y efecto que produce.

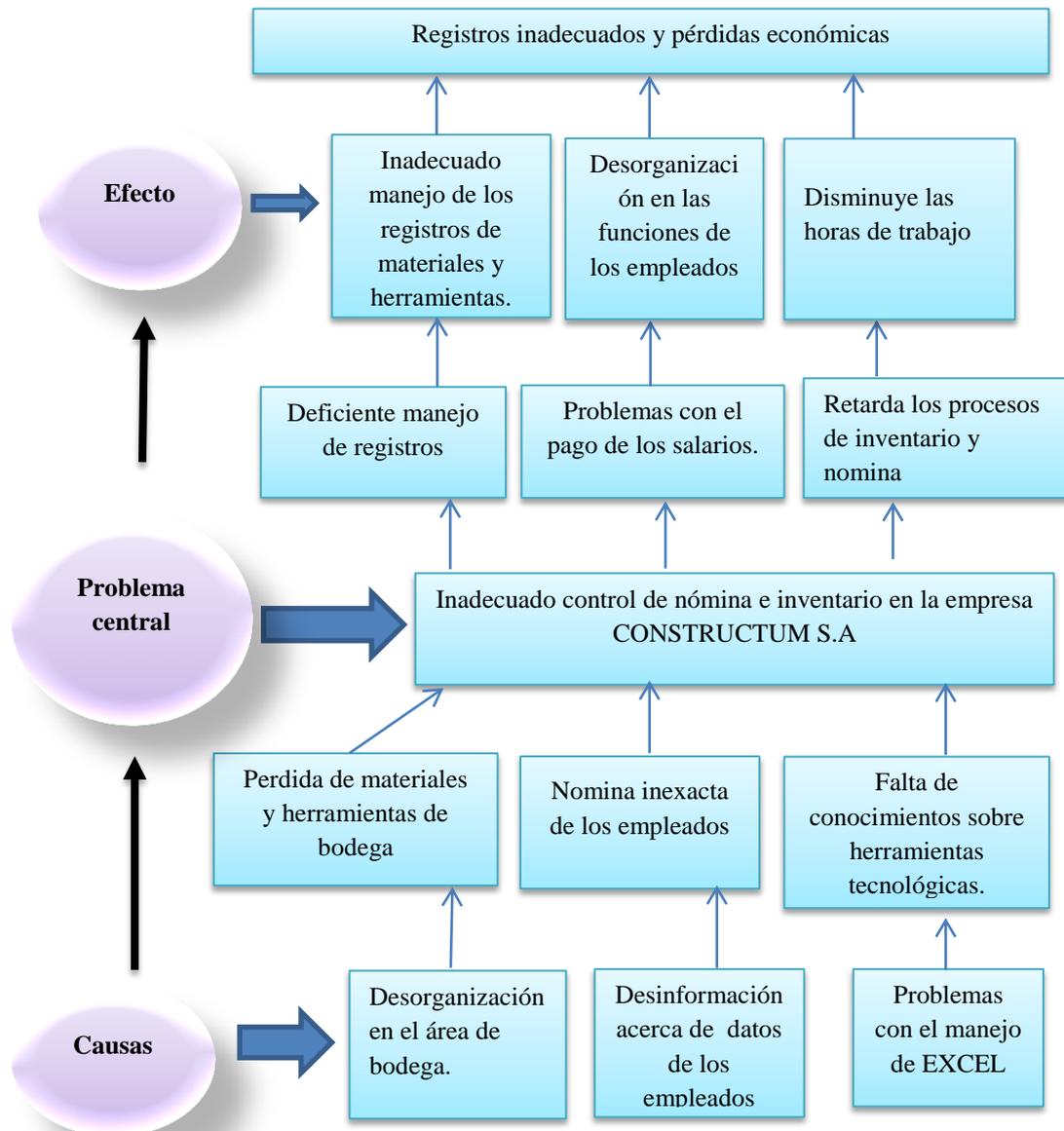


Figura 1: Árbol de problemas. Se utiliza para la identificación y organización de las causas y consecuencias del problema en la empresa "CONSTRUCTUM S.A".

### 3.02. Árbol de objetivos

El árbol de análisis de objetivos se lo realiza partiendo del árbol de problemas y transformándolos en aspectos positivos, iniciando por el propósito del proyecto, concatenando los componentes hasta la finalidad del proyecto.

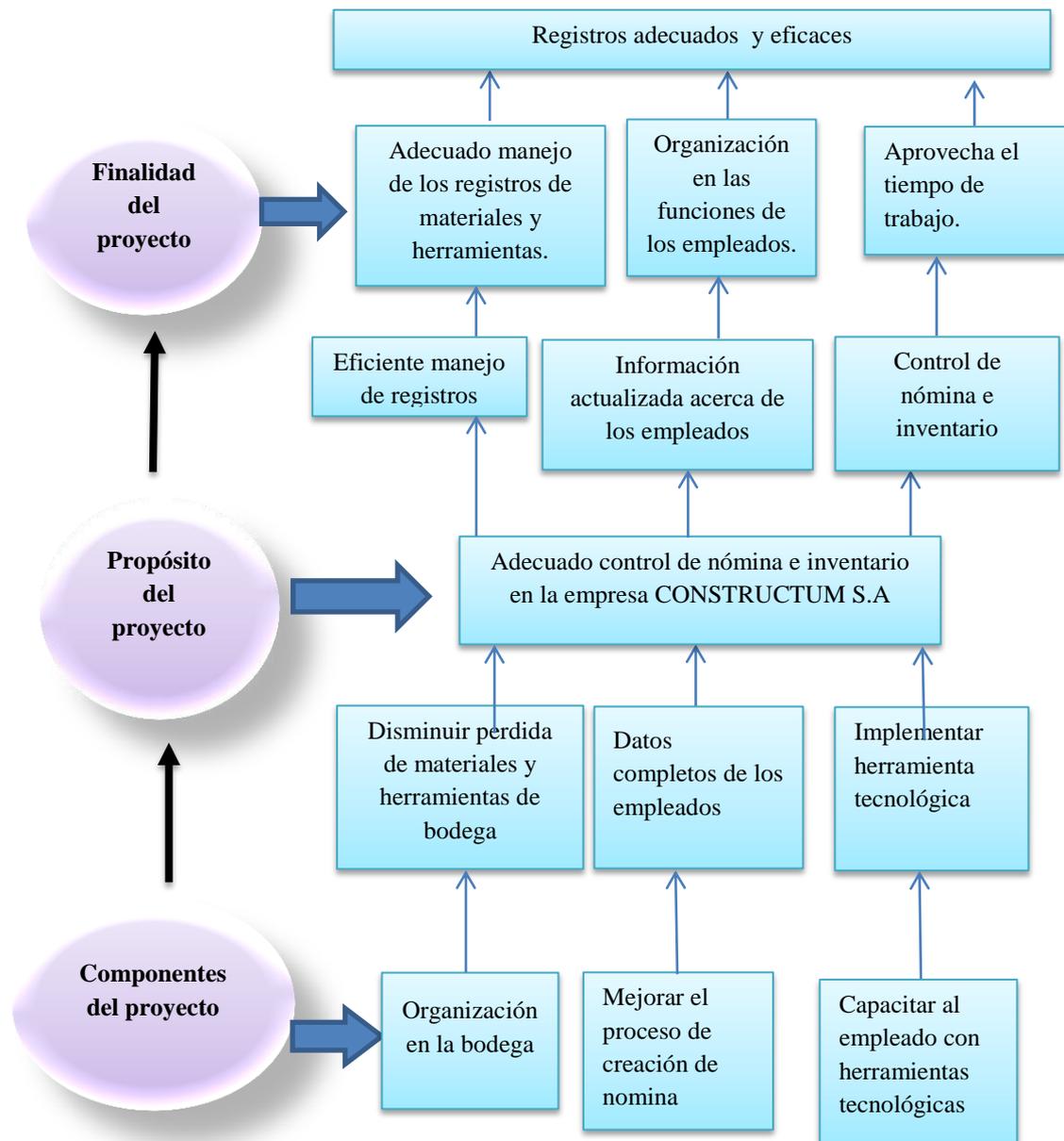


Figura 3: En el árbol de objetivos, los problemas encontrados en el árbol de problemas se convierten en soluciones o en objetivos.

### 3.03. Diagramas de casos de uso

En el siguiente diagrama explicaremos el proceso de nuestro sistema mediante casos de uso.

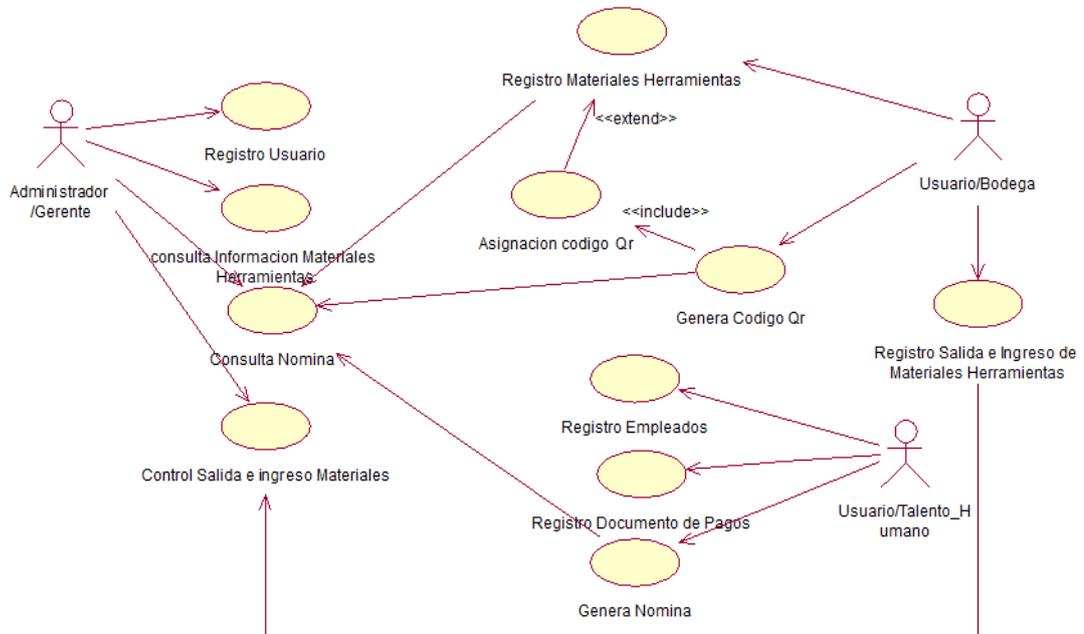


Figura 4: Diagrama de casos de uso general. Se representan los usuarios y a sus respectivos procesos que van a realizar dentro del sistema.

### 3.04. Casos de uso de realización

En un caso de uso de realización es una forma de que el usuario comprenda las acciones de los objetos. En los casos de realización se describen los objetos que colaboran su función.

**Caso: UC001**

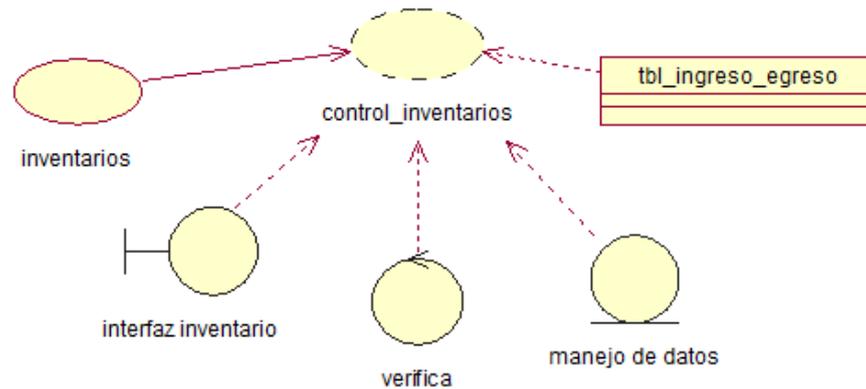


Figura 5: Diagrama de realización control de inventarios. En la figura se identifica los procesos para el control de inventario de bodega.

Tabla 14

*Especificación de diagrama de realización UCR001.*

<b>Nombre</b>	Control de inventario
<b>Identificador</b>	UCR001
<b>Responsabilidades</b>	Registro de información de bodega
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC001
<b>Casos de Uso</b>	
<b>Referencias</b>	RF001
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- El bodeguero tiene que estar registrado en el sistema.
	2.- Ingresar a interfaz bodega
<b>POSCONDICIONES</b>	
	1.- Solo el administrador y el bodeguero tendrán acceso a la interfaz de bodega.
	2.- La información deberá estar actualizada
<b>SALIDAS PANTALLA</b>	
	1.- Interfaz de bodega

**Caso: UC002**

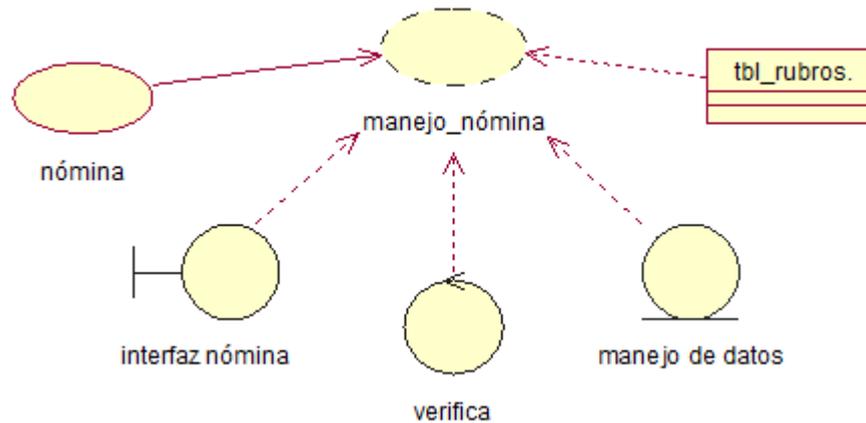


Figura 6: Diagrama de realización manejo de nómina. En la figura se identifica los procesos para el manejo de nómina.

Tabla 15

*Especificación de diagrama de realización UCR002.*

<b>Nombre</b>	Manejo de nómina
<b>Identificador</b>	UCR002
<b>Responsabilidades</b>	Generar nómina.
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC002
<b>Casos de Uso</b>	
<b>Referencias</b>	RF002
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- El usuario tendrá que estar registrado en el sistema. 2.- Ingresar a interfaz de nómina.
<b>POSCONDICIONES</b>	
	1.- Solo el administrador y el jefe de T.T.H.H tendrán acceso a la interfaz de bodega. 1.- Se logrará generar una nómina fiable de cada empleado de la empresa CONSTRUCTUM.
<b>SALIDAS PANTALLA</b>	
	1. Interfaz de T.T.H.H

**Caso: UC003**

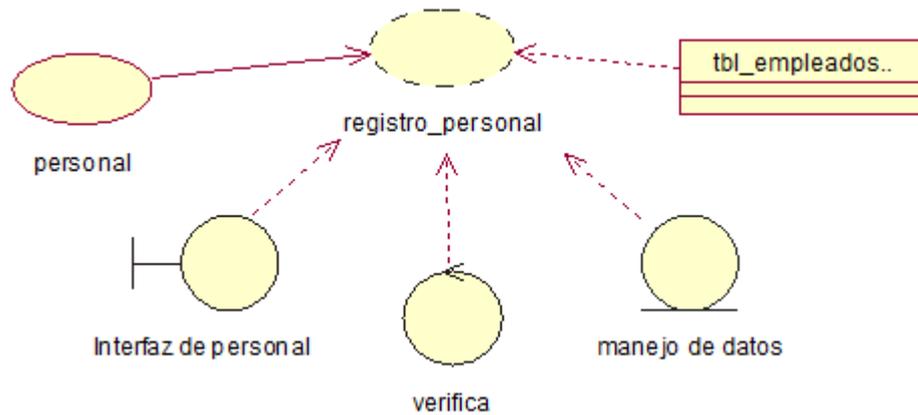


Figura 7: Diagrama de realización registro del personal. En la figura se identifica los procesos para el registro del personal en la base de datos del sistema.

Tabla 16

*Especificación de diagrama de realización registro del personal.*

<b>Nombre</b>	Registro del personal
<b>Identificador</b>	UCR003
<b>Responsabilidades</b>	Registrar a todos los empleados de la empresa CONSTRUCTUM
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC003
<b>Casos de Uso</b>	
<b>Referencias</b>	RF003
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- Ingresar al sistema con el rol de Administrador
	2.- El administrador deberá ingresar con su respectivo login y clave de usuario
<b>POSCONDICIONES</b>	
	1.- Únicamente el Administrador tendrá acceso.
	2.- Registrar y actualizar la información de cada empleado de la empresa CONSTRUCTUM
<b>SALIDAS PANTALLA</b>	
	1.- Interfaz personal

**Caso: UC004**

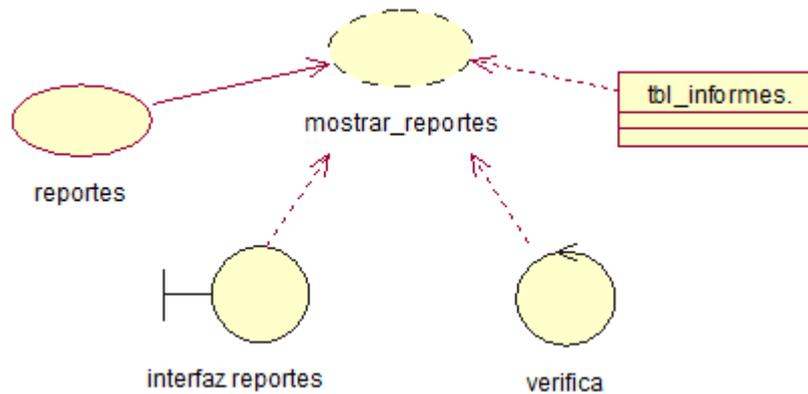


Figura 8: Diagrama de realización mostrar reportes. En la figura se identifica los procesos que se realizan al momento de solicitar un reporte.

Tabla 17

*Especificación de diagrama de realización UCR004*

<b>Nombre</b>	Mostrar reportes
<b>Identificador</b>	UCR004
<b>Responsabilidades</b>	Informar detalladamente los materiales y herramientas
<b>Tipo</b>	Sistema
<b>Referencias Casos de Uso</b>	UC004
<b>Referencias Requisitos</b>	RF004
<b>PRECONDICIONES</b>	
1.- Ingresar al sistema como Administrador o Bodeguero 2.- El usuario deberá tener privilegio para poder realizar una consulta 3.- El usuario debe solicitar en la búsqueda del sistema el reporte de material o herramienta	
<b>POSCONDICIONES</b>	
1.- El ingreso deberá estar habilitado para el Administrador y Bodeguero 2.- El usuario deberá solicitar el reporte de deseado sea material o herramienta	
<b>SALIDAS PANTALLA</b>	
1.- Interfaz reportes la cual nos indicará detalladamente la información del material o herramienta solicitado.	

**Caso: UC005**

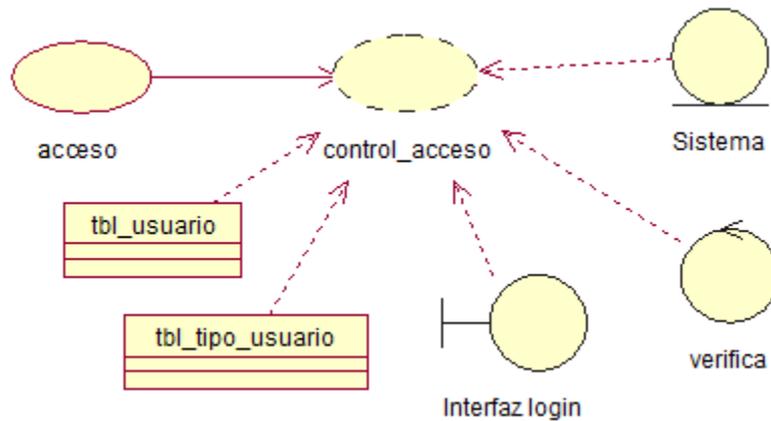


Figura 9: Diagrama de realización control de acceso. Representa lo que transcurre durante el proceso que realizan los usuarios al acceder al sistema

Tabla 18

*Especificación de diagrama de realización UCR005.*

<b>Nombre</b>	Control de acceso
<b>Identificador</b>	UCR005
<b>Responsabilidades</b>	Dar seguridad a la información
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC005
<b>Casos de Uso</b>	
<b>Referencias</b>	RF005
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- Estar registrado en el sistema de la empresa.
	2.- Tener un Login y password.
	3.- Ingresar al sistema
<b>POSCONDICIONES</b>	
	1.- El sistema deberá proporcionar el acceso y privilegios según el rol del usuario.
	2.- Si la clave es ingresada erradamente en 3 ocasiones se deberá bloquear por 15 min.
<b>SALIDAS PANTALLA</b>	
	1.- Interfaz del sistema

**Caso: UC006**

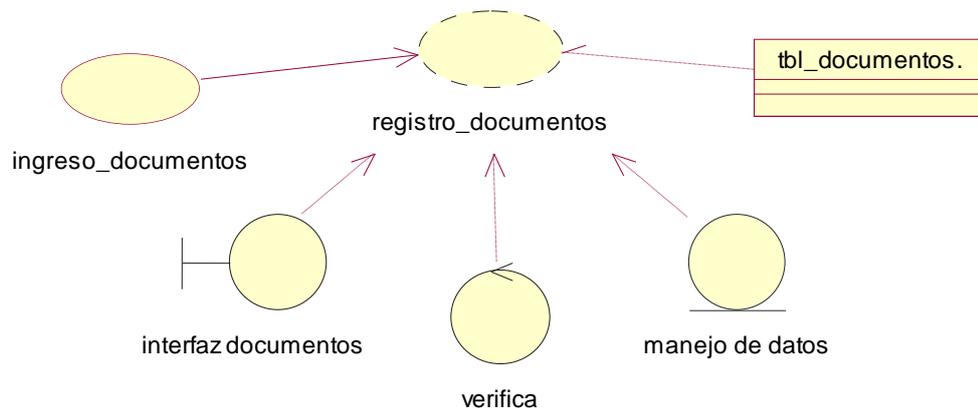


Figura 10: Diagrama de realización registro de documentos. Se representa al proceso del usuario al guardar la documentación de pagos de bodega

Tabla 19

*Especificación de diagrama de realización UCR006*

<b>Nombre</b>	Registro de documentos
<b>Identificador</b>	UCR006
<b>Responsabilidades</b>	Reportes
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC006
<b>Casos de Uso</b>	
<b>Referencias</b>	RF006
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- Ingresar al sistema con el rol de T.T.H.H
<b>POSCONDICIONES</b>	
	1.- Permitir acceso solo al usuario de T.H.H.H.
<b>SALIDAS PANTALLA</b>	
	1.- Interfaz documentos

**Caso: UC007**

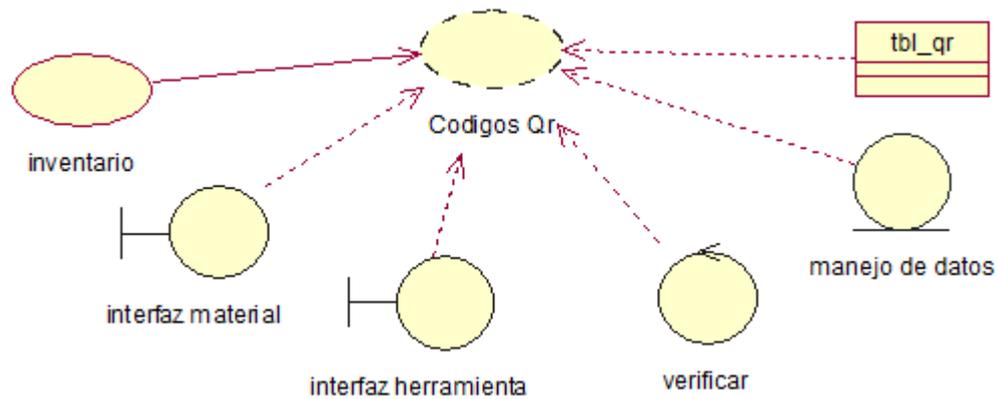


Figura 11: Diagrama de realización código Qr. En la figura se identifica los procesos del código Qr

Tabla 20

*Especificación de diagrama de realización UCR007.*

<b>Nombre</b>	Códigos Qr
<b>Identificador</b>	UCR007
<b>Responsabilidades</b>	Escanear códigos Qr
<b>Tipo</b>	Sistema
<b>Referencias</b>	UC007
<b>Casos de Uso</b>	
<b>Referencias</b>	RF007
<b>Requisitos</b>	
<b>PRECONDICIONES</b>	
	1.- Ingresar al sistema con el rol de Administrador o Bodeguero. 2.- Tener un registro mínimo de material o herramienta. 3.- Tener una webcam
<b>POSCONDICIONES</b>	
	1.- El sistema después de escanear el código Qr deberá indicarnos el reporte del material o herramienta.
<b>SALIDAS PANTALLA</b>	
	1.- Interfaz de Bodega

### 3.05. Diagrama de secuencias del sistema

El diagrama de secuencia es una manera gráfica y ordenada de demostrar los procedimientos q debe cumplir un objeto. En conclusión es como una secuencia de varios pasos y esto nos sirve para descubrir que objetos son necesarios para que puedan seguir dichos pasos.

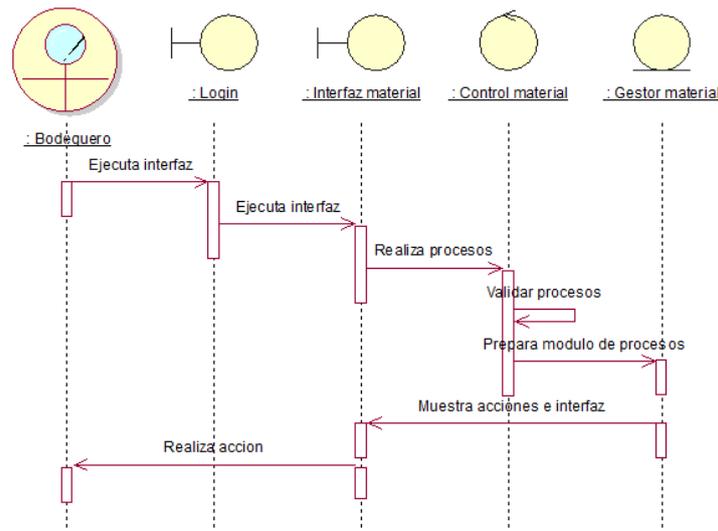


Figura 12: Diagrama de secuencia de control de inventario. En la figura se puede observar los procedimientos que se realiza cuando el bodeguero va a realizar un control de inventario.

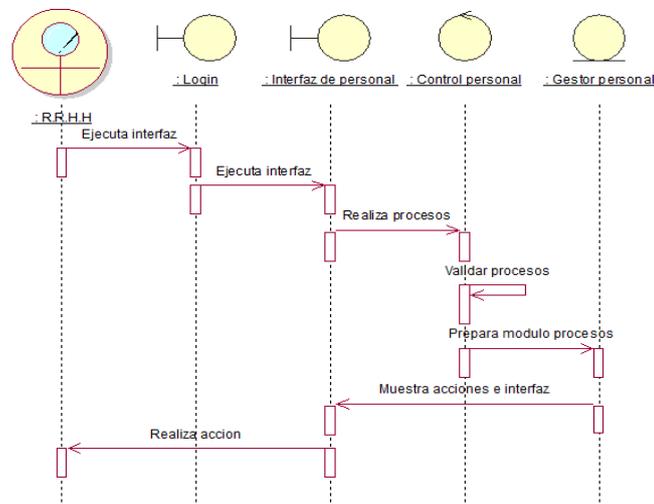


Figura 13: .Diagrama de secuencia manejo de nómina. En la figura se puede observar los procedimientos que se realiza cuando el usuario de T.T.H.H va a generar una nómina para el empleado.

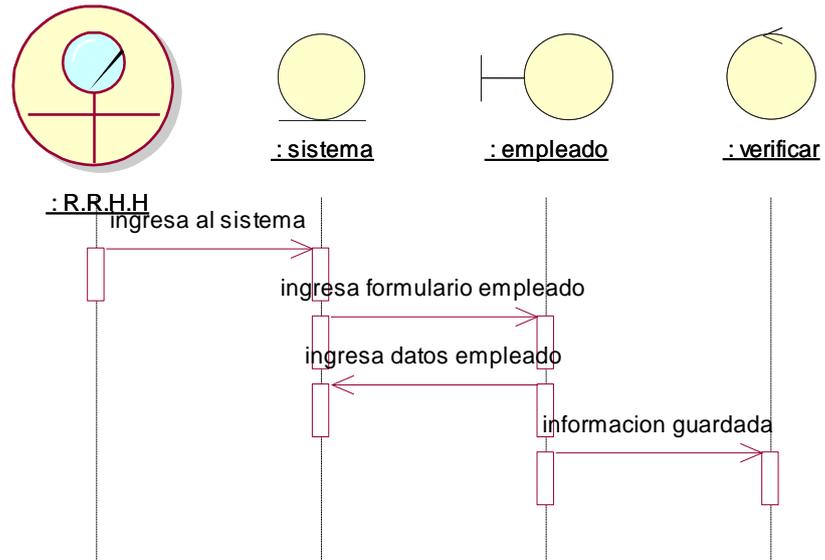


Figura 14: Diagrama de secuencia registro de personal. En la figura se puede observar el proceso del Administrador para poder actualizar la lista de empleados.

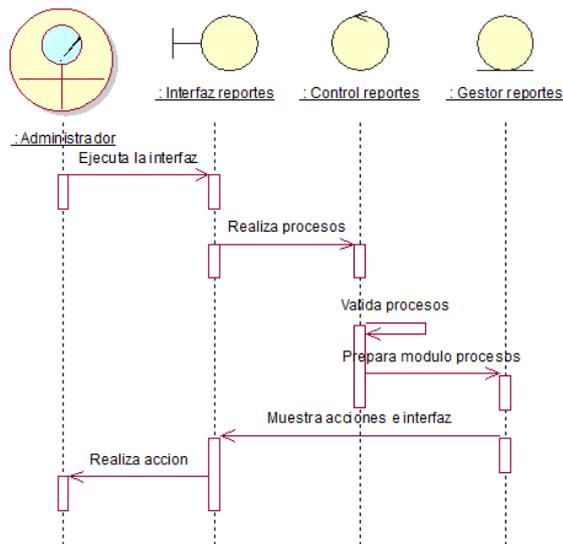


Figura 15: Diagrama de secuencia mostrar reporte. En la figura se puede observar el proceso del Administrador al momento de requerir un reporte.

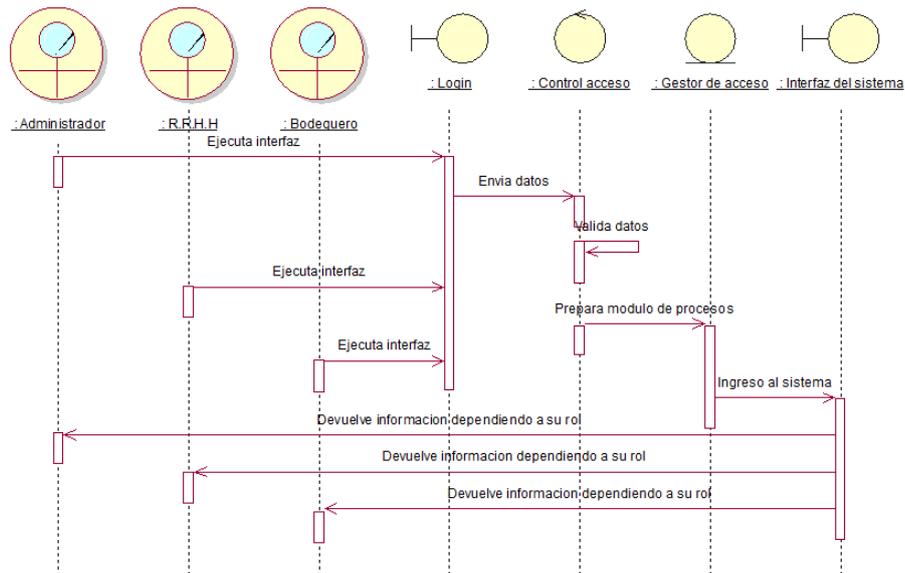


Figura 16: Diagrama de secuencia control de acceso. En la figura se puede observar el proceso de seguridad de acceso.

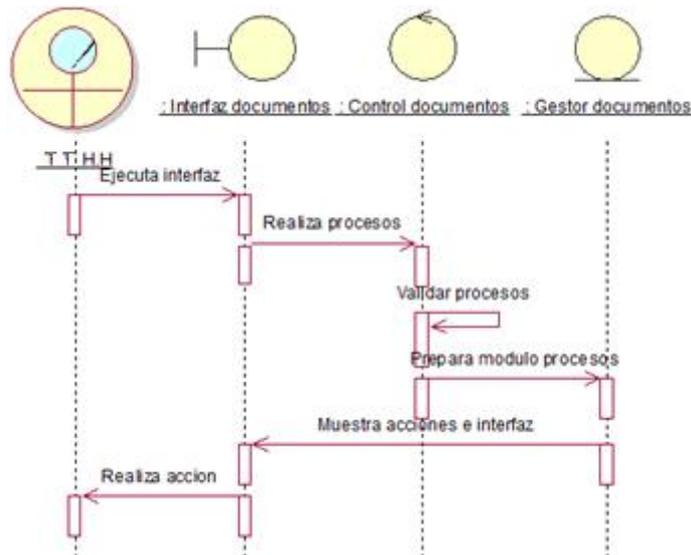


Figura 17: Diagrama de secuencia registro de documentos. En la figura se puede observar como el usuario de T.T.H.H realiza un registro de documentos de los pagos de bodega

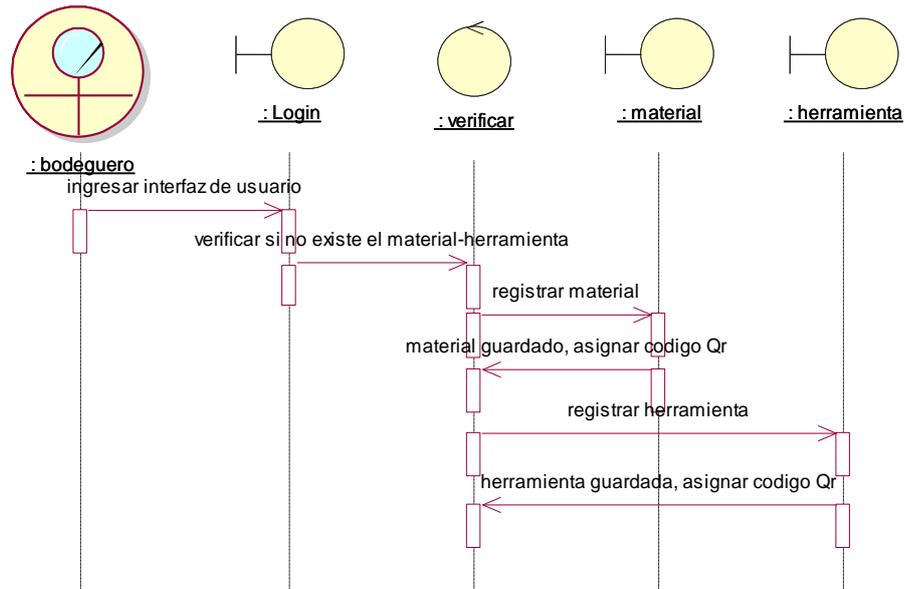


Figura 18: Diagrama de secuencia Códigos Qr. En la figura se indica el proceso de asignación a un código Qr

### 3.06. Especificación de Casos de Uso

La especificación de casos de uso nos sirve para identificar como el usuario interactúa con el sistema y también conocer sus delimitaciones.

Tabla 21

*Especificación de caso de uso UC001.*

<b>Caso de uso</b>	Bodega
<b>Identificador</b>	UC001
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario ingresa al sistema a la interfaz de bodega. Actualiza la información del inventario	El usuario podrá crear, modifica o elimina materiales y herramientas
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 22

*Especificación de caso de uso UC002.*

<b>Caso de uso</b>	Nómina
<b>Identificador</b>	UC002
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario T.T.H.H ingresa al sistema y genera la nómina de cada empleado.	Ingresa al sistema Ingreso a la Interfaz de documentos Ingresa el sueldo del empleado Devuelve el sueldo a pagar.
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 23

*Especificación de caso de uso UC003*

<b>Caso de uso</b>	Registro del personal
<b>Identificador</b>	UC003
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario T.T.H.H ingresa al sistema y actualiza la lista e información de los empleados.	Ingresa al sistema Ingreso a la Interfaz del personal Ingreso de información del empleado
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 24

*Especificación de caso de uso UC004*

<b>Caso de uso</b>	Mostrar Reportes
<b>Identificador</b>	UC004
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario ingresará al sistema ingresará a la interfaz bodega solicitará el material o herramienta.	El sistema deberá poder mostrar detalladamente los materiales o herramientas del inventario.
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 25

*Especificación de caso de uso UC005*

<b>Caso de uso</b>	Control de acceso
<b>Identificador</b>	UC005
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario ingresara su respectivo login y password	Permitir el acceso al sistema si coinciden el login y el password ingresados.
Ingresar al sistema	Ingresado 3 veces mal el login o el password se bloqueara por 15 min
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 26

*Especificación de caso de uso UC006*

<b>Caso de uso</b>	Registro de documentos
<b>Identificador</b>	UC006
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
El usuario de T.T.H.H deberá ingresar a la interfaz de documentos	El sistema deberá poder registrar la información de los pagos que son referentes a bodega
Registrar documentos	
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

Tabla 27

*Especificación de caso de uso UC007*

<b>Caso de uso</b>	Códigos Qr
<b>Identificador</b>	UC007
<b>Curso típico de Eventos</b>	
<b>Usuario</b>	Sistema
Ingresar al sistema	El sistema deberá permitir escanear códigos Qr y devolver como resultado el reporte del material o herramienta escaneado
Ingresar a la interfaz de bodega	
Escanear código Qr	
<b>Cursos Alternativos</b>	
1. El usuario podrá salir del sistema sin problemas	

## Capítulo IV: Análisis de alternativas

### 4.01. Matriz de Análisis de alternativas

En la siguiente matriz se identificara las soluciones alternativas que puedan llegar a ser estrategias para el proyecto y poder evaluar la factibilidad técnica, financiera, social y política permitiendo determinar la estrategia general y adaptarla al proyecto.

ISP= Impacto sobre el propósito, FT= Factibilidad Técnica, FF= Factibilidad Financiera, FS= Factibilidad Social, FP= Factibilidad Política, TOT= Total, CAT=Categoría.

Tabla 28

*Matriz de análisis de alternativas.*

MATRIZ DE ANÁLISIS DE ALTERNATIVAS							
OBJETIVOS	ISP	FT	FF	FS	FP	TOT	CAT
Organizar los registros de los departamentos involucrados	4	3	3	3	2	15	Media
Capacitar al personal para el uso del sistema	3	4	4	4	2	17	Media alta
Proporcionar reportes rápidos, concisos y apegados a la realidad.	4	4	2	3	2	15	Media
Evitar la pérdida de materiales-herramientas.	4	3	3	2	2	14	Media
Asegurar la información de la empresa y materiales-herramientas	4	3	2	4	2	15	Media
<b>TOTAL</b>	<b>19</b>	<b>17</b>	<b>16</b>	<b>16</b>	<b>10</b>	<b>76</b>	

#### 4.02. Matriz de Impactos de Objetivos

Esta matriz nos ayuda a entender y evaluar los cambios que genera en diferentes aspectos con respecto a los objetivos de nuestro proyecto.

Tabla 29

*Análisis del impacto de objetivos.*

<b>Objetivos</b>	<b>Factibilidad a lograrse</b>	<b>Impacto de genero</b>	<b>Impacto ambiental</b>	<b>Relevancia</b>	<b>Sostenibilidad</b>
Disminuir pérdidas económicas	•En factor costo beneficio, la mejora al avance de la constructora es evidente.	•En la constructora el personal femenino se encuentra muy conforme con la implementación del sistema	•Contribuye a proteger el entorno físico dentro y fuera de las instalaciones.	•Aclara las expectativas de los beneficiarios.	•Fortalece la Organización dentro de la constructora
Apoyo del gerente de la empresa para la aplicación del sistema.	•La constructora se auto financia a sí misma para su mejoramiento .	•El personal masculino se siente más seguro con el control de la bodega que contiene el sistema.	•Uso pleno del mejoramiento social.	•Es una precedencia apreciada por los favorecidos.	•Fortifica la colaboración del ámbito del negocio.
Mejorar la organización de registros, y fortalecer la seguridad de la información	• Es admisible y provechoso para la constructora.	Con el uso de herramienta no existe distinción de genero	•Protege los recursos naturales con uso de tecnología verde	•Los beneficios son aceptados por los clientes de la organización	•Hacer del sistema una herramienta diaria de trabajo.

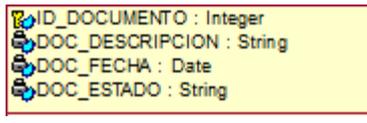
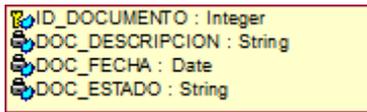
*Nota: Detalle de la matriz de impactos de objetivos*

#### 4.03. Estándares para el Diseño de Clases

Son notaciones para modelar sistemas orientados a objetos y describe la semántica esencial de lo que estos diagramas y símbolos significan.

Tabla 30

*Parámetros para el diseño del Diagrama de clases*

Nombre	Gráfico	Estándar
Clase	 <p>TBL_TIPO_EMPLEADO</p> <ul style="list-style-type: none"> <li>ID_TIPOEMPLEADO : Integer</li> <li>TEM_TIPO : String</li> <li>TEM_CARGO : String</li> <li>TEM_ESTADO : String</li> <li>CREAR()</li> <li>MODIFICAR()</li> <li>ELIMINAR()</li> </ul>	<p>El nombre de las clases serán en letras mayúsculas, iniciara con el sufijo "TBL" seguido de un guion bajo y el nombre de la clase.</p> <p>Ej.: "TBL_TIPO_EMPLEADO".</p>
Atributos	 <ul style="list-style-type: none"> <li>ID_DOCUMENTO : Integer</li> <li>DOC_DESCRIPCION : String</li> <li>DOC_FECHA : Date</li> <li>DOC_ESTADO : String</li> </ul>	<p>Se tomara las tres primeras letras de la clase seguida de un guion bajo con el nombre del atributo las letras serán en mayúsculas.</p> <p>Ej.: "DOC_DESCRIPCION".</p>
Métodos	 <ul style="list-style-type: none"> <li>CREAR()</li> <li>MODIFICAR()</li> <li>ELIMINAR()</li> </ul>	<p>Las letras serán en mayúsculas seguido de un paréntesis vacío.</p>
Primary Key	 <ul style="list-style-type: none"> <li>ID_DOCUMENTO : Integer</li> <li>DOC_DESCRIPCION : String</li> <li>DOC_FECHA : Date</li> <li>DOC_ESTADO : String</li> </ul>	<p>El identificador de cada tabla será el primary key.</p>
Relación de uno a muchos	<p>1 ——— *</p>	<p>Nos sirve para relacionar una tabla con otra y así obtener la herencia entre tablas.</p>
Dependencia o Instanciación (uso)	<p>-----&gt;</p>	<p>Denota la herencia de una tabla a otra.</p>

*Nota: Estándares del diseño de clases que se utilizó en el proyecto*

#### 4.04. Diagrama de Clases

Es una entidad, modela los conceptos del dominio de la aplicación, no es un software permite visualizar las relaciones entre las clases que involucran el sistema, un diagrama de clases está compuesto por: clases (atributos, operaciones y visibilidad) y relaciones (herencia, polimorfismo, encapsulamiento) (Ver A.02).

#### 4.05. Modelo Lógico – Físico

El modelo lógico de datos proporciona una visión general de los requisitos del negocio capturados ya que pertenecen a entidades de datos. El diagrama de modelo de datos representa gráficamente el modelo físico. El modelo de datos físico utiliza requerimientos capturados del modelo de datos lógicos (Ver A.03).

#### 4.06. Diagrama de Componentes

Un componente es una parte física de un sistema (módulo, base de datos, programa ejecutable, se puede decir que un componente es la materialización de una o más clases, porque una atracción con atributos y métodos pueden ser implementados en los componentes.

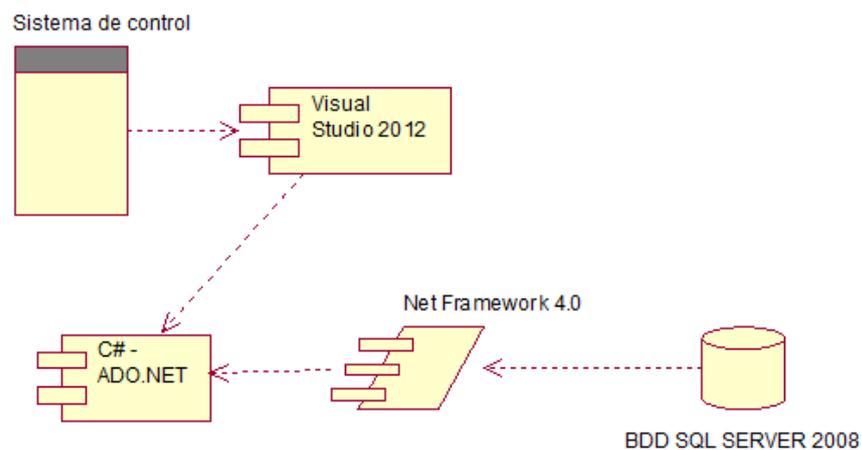


Figura 19: Diagrama de componentes del sistema. Indica la perspectiva intangible del proyecto.

#### 4.07. Diagrama de Estrategias

El diagrama de estrategias se define a partir de las cadenas de objetivos enlazados seleccionados, consignando los objetivos en sus niveles jerárquicos correspondientes

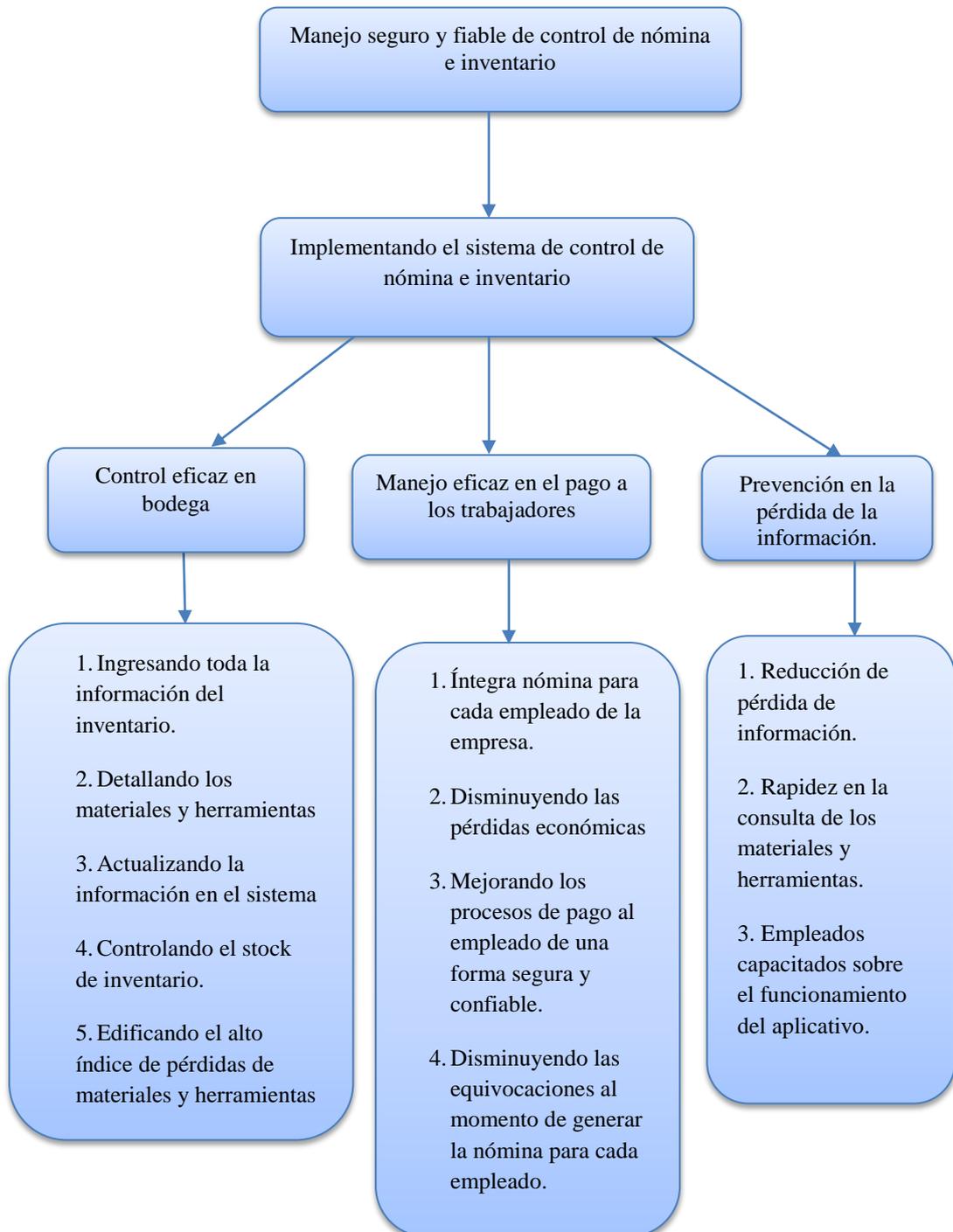


Figura 20: Detalle del diagrama de estrategias. Indica las soluciones del árbol de problemas para cumplir con el árbol de objetivos

#### 4.08. Matriz de Marco Lógico

El marco lógico es una herramienta para facilitar el proceso de conceptualización, diseño, ejecución y evaluación de proyectos. Su propósito es brindar estructura al proceso de planificación y comunicar información esencial relativa al proyecto.

Tabla 31

*Matriz marco lógico.*

Matriz Marco Lógico			
Resumen Narrativo	Indicadores	Medios de Verificación	Supuestos
Finalidad Controlar el inventario de bodega y mejorar la generación de nómina del empleado	1.-En la empresa CONSTRUCTUM S.A se verificará el funcionamiento del aplicativo en el 100% esperando satisfacer las necesidades y expectativas de los empleados.	Aceptación del Gerente de la empresa por medio del documento firmado.	El inventario se encuentra actualizado.
Propósito Registro de nóminas e inventarios confiables.	1.- Porcentaje de ventas subieron gracias al inventario completo y actualizado.	Uso de registros y reportes.	Se controla el stock de materiales y herramientas.
Componentes 1.- El registro de inventario es seguro.  2.- La nómina del empleado es integra.  3.- identificar los instrumentos que se necesitaran para la implementación del sistema.	1.-Información de los procesos actuales en el registro del inventario y nóminas.  2.- Software y hardware que se necesitan.  3. Porcentaje alto de inconsistencia de datos.	-Entrevista al gerente  -Revisión visual de los equipos.	-Funcionamiento completo del sistema.  -Infraestructura necesaria.
Actividades 1. Capacitación del personal.  2. Configuración del ambiente donde se implementara el sistema.	1. Capacitación al personal sobre el manejo del sistema.  2. Se compra un computador con el presupuesto de 500 dólares.	Conferencia dictada al Jefe de T.T.H.H y el jefe de bodega.  Instalación de programas.  Facturas.	El personal a capacitar se encuentra ausente.  Falta de presupuesto.

## 4.09. Vistas Arquitectónicas

### 4.09.01. Vista lógica

Apoya principalmente los requisitos funcionales, lo que el sistema debe brindar en términos de servicios a sus usuarios.

El sistema se descompone en una serie de abstracciones primarias, tomadas principalmente del dominio del problema en la forma de objetos o clases de objetos.

Aquí se aplican los principios de abstracción, encapsulación y herencia. Esta descomposición no solo se hace para potenciar el análisis funcional, sino también sirve para identificar mecanismos y elementos de diseño comunes a diversas partes del sistema.

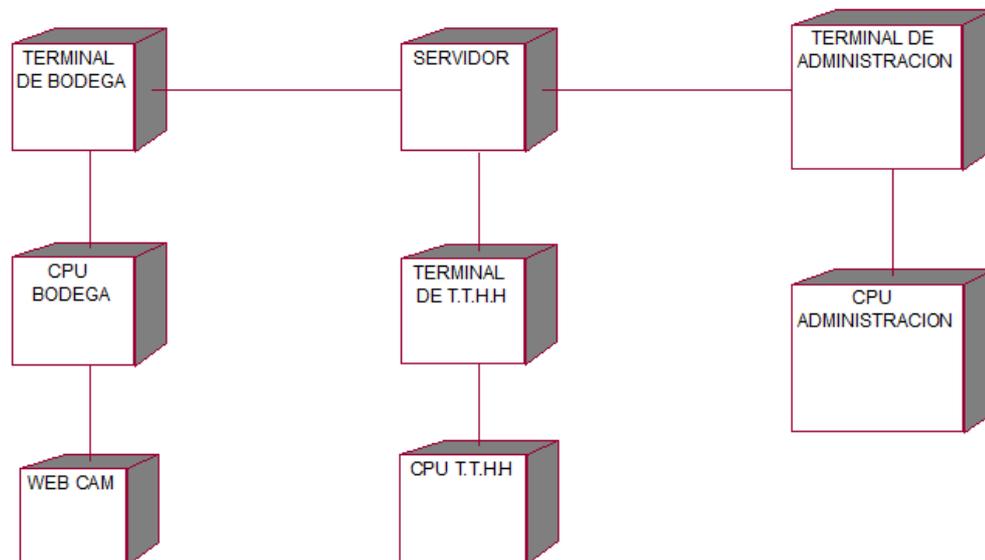


Figura 21: Detalle de la Vista lógica del sistema. Muestra los componentes lógicos del sistema

#### 4.09.02. Vista física

Se toma en cuenta los requisitos no funcionales del sistema tales como.

Disponibilidad, confiabilidad, desempeño entre otras. El sistema se ejecuta sobre varios nodos de procesamiento (hardware). Estos nodos son relacionados con elementos identificados de las vistas anteriores.

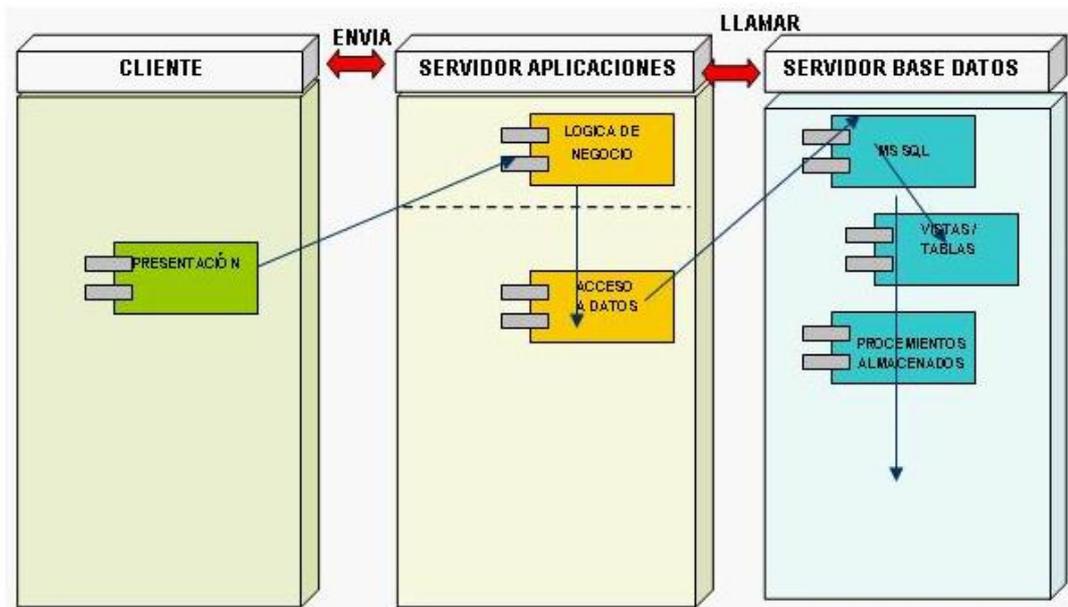


Figura 22: Detalle de la Vista física del sistema. El diagrama indica la vista física del sistema.

#### 4.09.03. Vista de desarrollo

Muestra los subsistemas que son desarrollados en una jerarquía de capas, cada una brinda una interfaz estrecha y bien definida hacia las capas superiores.

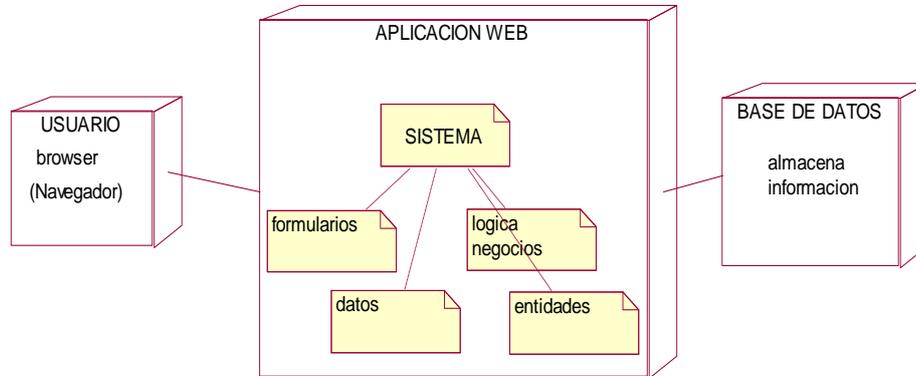


Figura 23: Detalle de la Vista de desarrollo. Indica la organización de los módulos del software sobre el entorno de desarrollo.

#### 4.09.04. Vista de procesos

Se tratan los aspectos de concurrencia y distribución, integridad del sistema y tolerancia a fallos. Se especifica en cual hilo de control se ejecuta efectivamente una operación de una clase identificada en la vista lógica.

Puede ser descrita como un conjunto de redes lógicas de procesos que son ejecutados de forma independiente, y distribuidos a lo largo de varios recursos de hardware conectados mediante un bus o a una red de datos.

## Diagramas de actividades

El diagrama de actividades se utiliza para mostrar una visión simplificada de lo que ocurre durante una operación o proceso.

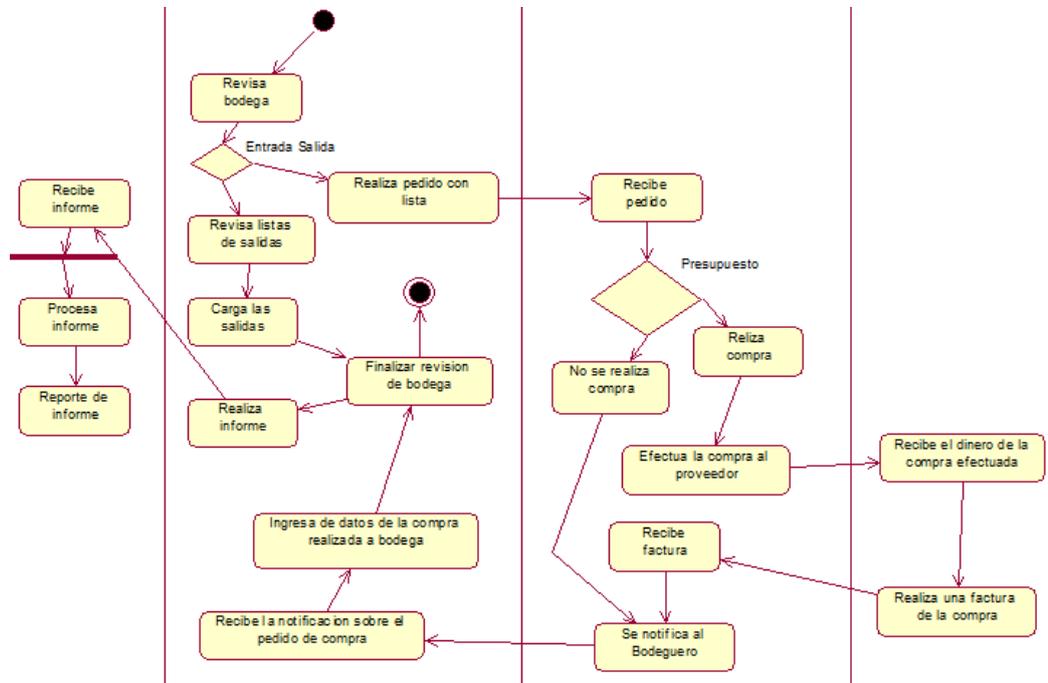


Figura 24: Detalle de actividades de bodega. La figura indica las actividades que transcurren en los procesos de bodega.

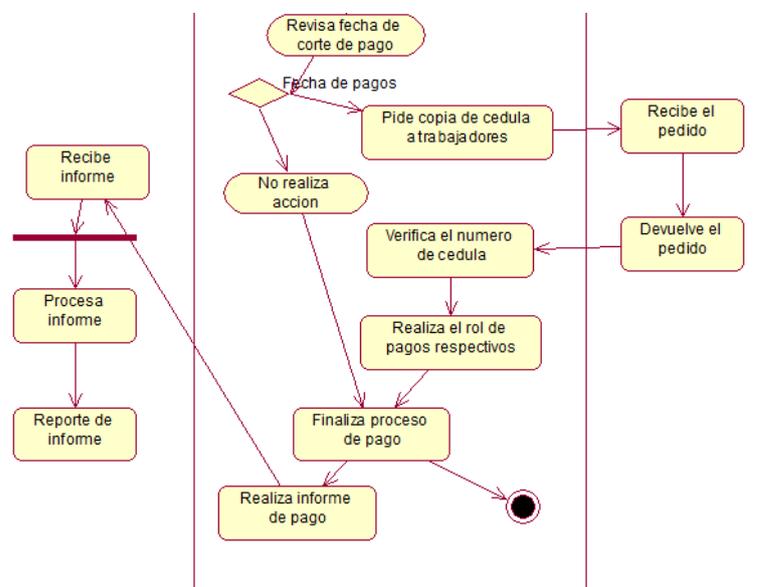


Figura 25: Detalle del diagrama de nómina. La figura indica las actividades que transcurren en los procesos de pagos del empleado.

## Capítulo V: Propuesta

### 5.01. Especificación de estándares de programación

Se describe los estándares que se utilizaron para la creación del código fuente del sistema (Ver A.04).

Tabla 1

Controles.

Tipo de control	Prefijo	Ejemplo
Label	Lbl	Lbl_titulo
TextBox	Txt	Txt_nombre
Button	Btn	Btn_aceptar
RadioButton	Rdo	Rdo_imagen
CheckBox	Chk	Chk_estado
DropDownList	Cmb	Cmb_rol
Table	Tbl	Tbl_cliente

*Nota: Detalle de las normas de tipo control que se emplearon para la creación del código fuente.*

Tabla 2

Tipo de Datos.

Tipo de variable	Mnemónico	Descripción
Char	Ch	Un carácter UNICODE de 16 bits
String	St	Cadena de caracteres
Integer	In	Entero de 32 bits con signo.
Date	Dt	Formato de fecha/hora
Boolean	Bl	Valor lógico: verdadero y falso
Float	Fl	Coma flotantes, 11-12 dígitos significativos.
Double	Db	Coma flotante, 64 bits (15-16 dígitos significativos)
Object	Ob	Objeto genérico
Byte	By	Entero de 8 bits sin signo.
Array	Arr	Una matriz es un tipo de datos compuesto que puede contener valores de datos múltiples

*Nota: Detalle de las normas de tipo de datos que se emplearon para la creación del código fuente*

## 5.02. Diseño de Interfaces de Usuario

Se muestra la interfaz de usuario



Figura 26: Interfaz del sistema.

En esta ventana se puede observar las opciones:

A= Logo de la empresa

B= Submenú del sistema

C= Bodega (Se realiza el mantenimiento de material al igual que el de herramienta los cuales se podrán ingresar al sistema)

D= Administración (El mantenimiento de cada uno de los usuarios que podrán trabajar con el sistema).

E= Personal (se ejecuta el pago de personal, así como el mantenimiento de los trabajadores del negocio).

f= Documentos (Ingreso, modificación y eliminación de documentos del negocio).

## Personal



Figura 27: Interfaz del personal.

A= Al realizar click en nuevo se desplegara otra pantalla para poder ingresar los datos del trabajador.

D = Al escoger un criterio en el combo box se logrará buscar un trabajador ya sea por apellido o código. Al realizar click en buscar se mostrara el resultado.

B= En cada uno de los trabajadores se podrá observar que por su delante tiene un esfero que sirve para editar sus datos

C= El símbolo equis permite eliminar al trabajador.

## Pantalla para ingresar nuevo trabajador o editar sus datos

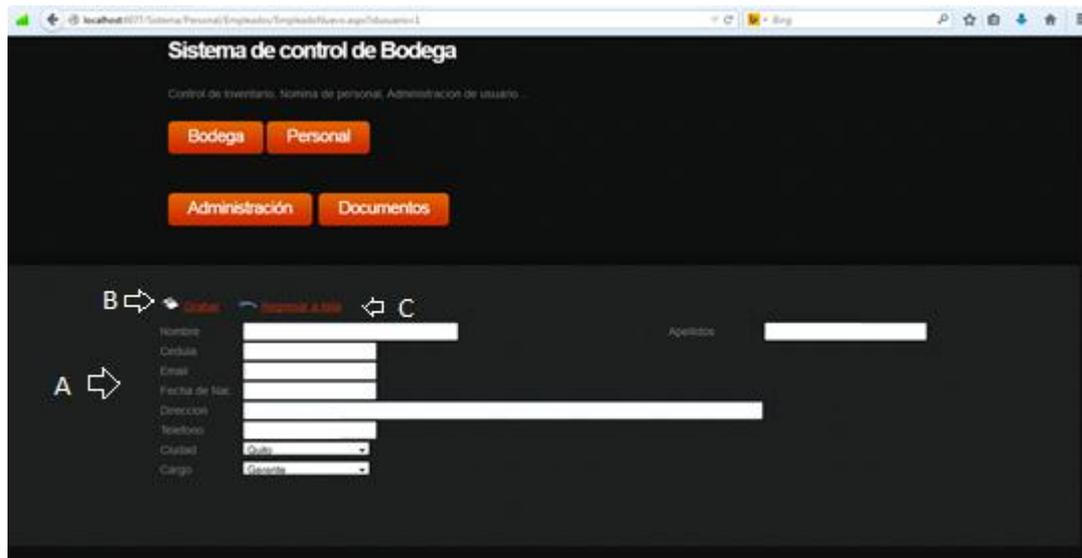


Figura 28: Interfaz del personal.

En esta pantalla observaremos los datos a editar del trabajador o simplemente un ingreso de datos para un nuevo trabajador.

A= Campos a llenar de información del trabajador

B= Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo trabajador o una edición del mismo.

C=Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, que ara el sistema es regresarnos al menú de personal.

### 5.03. Especificación de pruebas de unidad

Comprueba si el diseño es fiable acorde a las relaciones existentes entre las tablas de la base datos. Enfocándose en la información que almacena la tablas del sistema.

Tabla 33

*Especificación de la prueba de unidad PUD001.*

Identificador de la Prueba	PDU001
Método a Probar	Validación del Usuario
Objetivo de la Prueba	Demostrar que el ingreso al sistema tiene seguridad.
Datos de Entrada	
1.-Login	
2.- Password	
Resultado Esperado	
1.- Si el usuario se encontraba registrado en el sistema con su respectivo login y password accederá al sistema sin ningún inconveniente caso contrario al no estar registrado se denegara el acceso.	
Comentario	
1.- Si se ingresa 3 veces mal el login y el password el sistema se bloqueara por 15 minutos.	

Tabla 34

*Especificación de la prueba de unidad PUD002*

Identificador de la Prueba	PDU002
Método a Probar	Ingreso, Modificación; Eliminación de materiales y herramientas del inventario
Objetivo de la Prueba	Demostrar que el manejo de inventario se realiza eficazmente
Datos de Entrada	
Información de material y herramienta	
Resultado Esperado	
1.- Funcionamiento fiable del manejo de inventario en el sistema	
Comentario	
1.- El registro de material será diferente al de herramienta.	

#### 5.04. Especificación de pruebas de aceptación

Tabla 35

*Especificación de la prueba de aceptación EPA001.*

Identificador de la Prueba	EPA001
Caso de uso	Ninguno
Tipo de usuario	Administrador
Objetivo de la Prueba	Registrar usuarios que utilizaran el sistema
Secuencia de Eventos	1.- Asignar el rol del usuario en el sistema 2.- Asignar un login y contraseña
Resultado Esperado	1.- Usuarios registrados exitosamente 2.- El sistema se habilitara acorde a las necesidades del rol del usuario
Comentario	Ninguno
Estado:	Aceptado

Tabla 36

*Especificación de la prueba de aceptación EPA002.*

Identificador de la Prueba	EPA002
Caso de uso	UC001
Tipo de usuario	Gerente
Objetivo de la Prueba	Ingresar ,Modificar, Eliminar Materiales exitosamente
Secuencia de Eventos	1.-Registrar , modificar ,eliminar material 2.- Proceso realizado con éxito
Resultado Esperado	1.- El sistema cumplirá las funciones de acuerdo al tipo de usuario
Comentario	Ninguno
Estado:	Aceptado

## 5.05. Especificación de pruebas de carga

Tabla 37

*Especificación de la prueba de carga PDCD001.*

Identificador de la Prueba	PDC001
Tipo de usuario	Administrador
Objetivo de la Prueba	Manejar inventario con dos o más usuarios conectados
Descripción	Realizar procesos de inventario cuando se encuentran conectados dos o más usuarios al sistema
Resultado Esperado	Estabilidad del sistema.
Comentario	El sistema funciono correctamente al tener 2 o más usuarios conectados

## 5.06. Configuración del Ambiente mínima/ideal

Las siguientes recomendaciones son ideales para tener un correcto funcionamiento de un sistema orientado a la web y alcanzando con esto una implementación del sistema óptima que darán la satisfacción de los usuarios.

### Requerimientos de Hardware

#### Equipo Servidor

- Procesador : Intel Core 2 Duo de 2 Ghz ó 3Ghz
- Memoria Ram 4 GB.Dsco Duro con 100 Gb LIBRES para datos.

#### Equipo Cliente

- Procesador: Intel Core 2 Duo de 2GHz.
- Memoria Ram 2 Gb.
- Disco Duro: 100 Gb

## **Requerimientos de Software**

Visual Studio 2012

SQL SERVER R2

## **Equipo Cliente**

Sistema Operativo

- Microsoft Windows 7 o superior.

Browser

- Mozilla Firefox
- Google Chrome

## Capítulo VI: Aspectos administrativos

### 6.01. Recursos

Tabla 38

*Recurso Humano.*

RECURSOS HUMANOS			
Humano	Nombre	Actividad	Responsabilidad
Promotor del proyecto	Bryan Peñafiel	Desarrollo del proyecto.	Desarrollar el sistema y la documentación.
Tutor	Ing. Hugo Heredia	Tutelar el desarrollo del proyecto.	Analizar el progreso del desarrollo del sistema y la documentación.
Lector	Ing. Patricia Garzón	Examinar el proyecto.	Asegurar que el desarrollo del proyecto elaborado cumple con las normas establecidas.

Tabla 39

*Recurso Material.*

RECURSOS MATERIALES			
Material	Cantidad	Valor Unitario	Valor Total
Computador	1	800	800
Hojas de papel bond	600	0,05	30
Esfero	2	0,35	0,70
Lápiz	2	0,35	0,70
Borrador	2	0,15	0,30
Alimentación	120 días	2,00	240,00
Transporte	120 días	1,50	180,00
Total			1.251,70



## Capítulo VII: Conclusiones y recomendaciones

### 7.01. Conclusiones

- El sistema compensa las necesidades de los empleados de la empresa Constructum S.A en los proceso de control de inventario y nómina asegurando la integridad y seguridad de la información, optimizando el registro actualizado de material o herramienta de bodega y generando una confiable nómina a cada empleado de la empresa.
- Al aplicar el sistema en el área de control de inventarios ha reducido el alto índice de pérdidas de material y herramienta, el bodeguero al registrar toda la información en el sistema tener un control del stock en bodega.
- El software ha logrado brindar eficacia al proceso de obtención de informes con respecto al inventario de materiales y nómina de empleados, ofreciendo información mucho más confiable.
- Se ha logrado mejorar el mecanismo de recolección de información tanto en bodega como en el departamento de recursos humanos, obteniendo como beneficio seguridad de información, y digitalización de la información.
- Definimos roles de usuario proporcionando accesibilidad al sistema de acuerdo con las necesidades de cada uno.

## 7.02. Recomendaciones

- Capacitaciones a cada uno de los usuarios que intervienen en el manejo del software el cual mantendrá al personal actualizado acerca de los procesos de registro y la utilización del mismo.
- Manuales de usuario a disposición de los empleados que manipulan el sistema.
- Uno de los requisitos en el perfil del empleado que tenga acceso a la aplicación sea obtener conocimientos básicos sobre informática.
- El sistema será únicamente utilizado por el personal autorizado en el registro y control de materiales-herramientas y nóminas de usuario.
- Manual técnico que servirá de respaldo para una actualización en el software.



# Anexos

## A.0 1

## Descripción detallada de la matriz de requerimientos

**Matriz de requerimientos**

Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
<b>REQUERIMIENTOS FUNCIONALES</b>						
RF001	El sistema debe permitir manejar el inventario	Administrador Bodega	Alta	Software	En revisión	Administrador Bodeguero
RF002	El sistema debe poder generar la nómina del empleado	Administrador Bodega	Alta	Software	En revisión	Administrador T.T.H.H.
RF003	El sistema podrá registrar los datos del personal	Administrador T.T.H.H	Alta	Software	En revisión	Administrador
RF004	El sistema deberá mostrar reportes de cada material y herramienta del inventario	Administrador	Alta	Software	En revisión	Administración Bodeguero
RF005	El sistema debe restringir el acceso según el rol y el cargo	Administrador	Alta	Software	En revisión	Administración T.T.H.H Bodeguero
RF006	El sistema deberá permitir registrar la documentación referente a los pagos de bodega.	Bodeguero	Alta	Software	En revisión	Bodeguero
RF007	El sistema debe permitir la creación y escaneo de códigos Qr.	Administrador	Alta	Software	En revisión	Administrador Bodeguero
RNF001	El sistema debe ser de fácil navegabilidad.	Administrador	Media	Usuario	En revisión	Administrador T.T.H.H Bodeguero



---

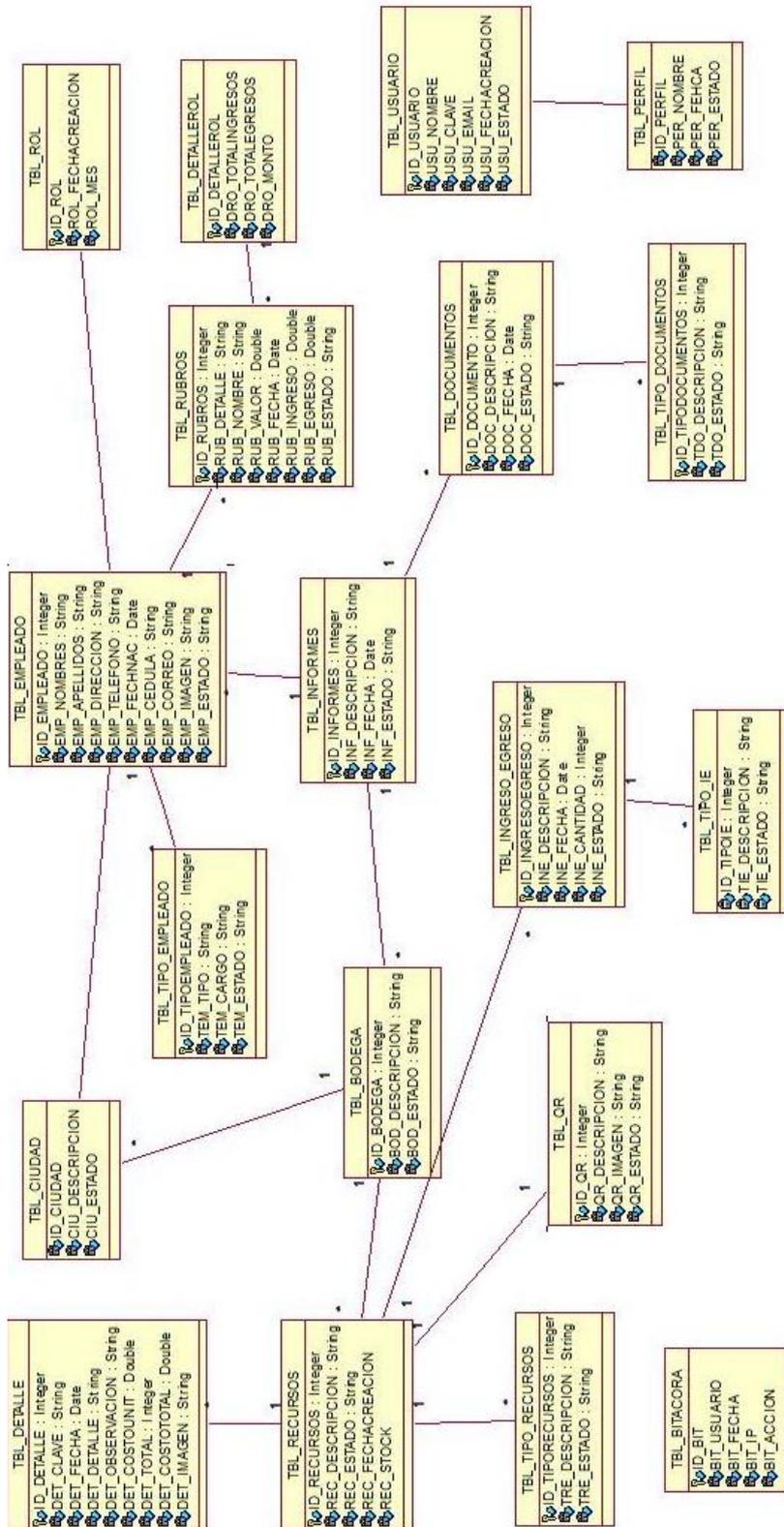
RNF002	El sistema debe ser seguro y mantener la integridad de la información.	Administrador	Media	Usuario	En revisión	Administrador T.T.H.H Bodeguero
RNF003	El sistema debe controlar el acceso a los diferentes módulos.	Administrador	Media	Usuario	En revisión	Administrador T.T.H.H Bodeguero

---



A.03

Modelo Físico-Lógico





A.0 4

Parámetros de la programación.

NOMBRE	GRAFICO	DESCRIPCIÓN
<b>Código de declaración ASPX</b>	<pre>&lt;asp:DataGrid ID="Dgr_HerramientasLista" runat="server"   &lt;AlternatingItemStyle BackColor="white" /&gt;   &lt;EditItemStyle BackColor="#2461BF" /&gt;</pre>	El código de declaración de una etiqueta ASPX
<b>Literal de librerías de las capas.</b>	<pre>using System.Collections.Generic; using System.Linq; using System.Web; using System.Web.UI; using System.Web.UI.WebControls; using Acceso_Datos; using Logica_Negocios; using System.Data.Linq;</pre>	Las librerías se utilizan para poder realizar instancias a las capas de Base de datos y Lógica de Negocio
<b>Literales de cadena</b>	<pre>string aux="Bodega";</pre>	Cuando una cadena es literal (no contiene sustitución de variables), el apóstrofe o "comilla simple" siempre se deben utilizar para delimitar la cadena:
<b>Instancia a la Base de datos</b>	<pre>public static DB_SISTEMADataContext dc = new DB_SISTEMADataContext();</pre>	Instancia en una variable de toda la base de datos para poder controlar Tablas y campos de su respectiva tabla.
<b>Métodos públicos</b>	<pre>public static List&lt;TBL_USUARIO&gt; Lista_usuario() {   if   {     var lista = dc.TBL_USUARIO.Where(usu =&gt; usu.USU_ESTADO.Equals("A"));     return lista.ToList();   } }</pre>	Método público que devuelve una consulta directa a la Base de datos
<b>Método públicos con Constructor</b>	<pre>public static bool ingreso(string nombre, string clave) {   var existe = dc.TBL_USUARIO.Any(exi =&gt; exi.USU_NOMBRE.Equals(nombre) &amp; exi.USI   return existe; }</pre>	Método público que devuelve una consulta a través de parámetros, directa a la Base de datos
<b>Métodos privados</b>	<pre>private void CargarUsuario() {   userList = Logica_Recursos.Lista_recursos_herrxBodega(codigo)   if (userList != null)   {     Dgr_HerramientasLista.DataSource = userList;     Dgr_HerramientasLista.DataBind();   } }</pre>	Métodos internos de cada página web que aplica la programación orientada objetos, en la reutilización de código, también puede ir con constructor.
<b>Instancia a las tablas de la Base de Datos</b>	<pre>TBL_RECURSOS userinfo = new TBL_RECURSOS(); List&lt;TBL_RECURSOS&gt; userList = new List&lt;TBL_RECURSOS&gt;();</pre>	Instancia a las tablas para poder realizar consultas a la tabla o realizar inserción de registros para mas adelante poder guardarlo en la base de datos

---

**Registros**

```
// save
public static void save(TBL_USUARIO usuarioInfo)
{
    try
    {
        usuarioInfo.ID_USUARIO = IdCodigo();
        usuarioInfo.USU_ESTADO = "A";
        //validacion(usuarioInfo);
        dc.TBL_USUARIO.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO GUARDADOS <br>" + ex.Message);
    }
}
```

Inserción de datos enviados desde el sistema a las tablas del sistema de la Base de datos.

---

**Archivos js**

```
(function ($) {
    $('#document').ready(function () {
        $('#qrcodebox').WebcamQRCode({
            onQRCodeDecode: function (p_data) {
                //$('#qrcode_result').html(p_data);
                $('#<del>qrcode_result.ClientID</del>').val(p_data);
            }
        });
    });

    $('#btn_start').click(function () {
        $('#qrcodebox').WebcamQRCode().start();
    });

    $('#btn_stop').click(function () {
        $('#qrcodebox').WebcamQRCode().stop();
    });
})(jQuery);
</script>
```

---

**Lo archivos js deberán estar dentro de las etiquetas <script></script> .**

A.0 5

Script de la base de datos

```
USE [DB_SISTEMA]
```

```
GO
```

```
/****** Object: Table [dbo].[TBL_TIPO_RECURSOS] Script Date: 10/11/2014
```

```
18:55:29 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_TIPO_RECURSOS](
```

```
    [ID_TIPORECURSOS] [int] NOT NULL,
```

```
    [TRE_DESCRIPCION] [varchar](50) NULL,
```

```
    [TRE_ESTADO] [varchar](50) NULL,
```

```
    CONSTRAINT [PK_TBL_TIPO_RECURSOS] PRIMARY KEY CLUSTERED
```

```
(
```

```
    [ID_TIPORECURSOS] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/****** Object: Table [dbo].[TBL_TIPO_IE] Script Date: 10/11/2014 18:55:29
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_TIPO_IE](
```

```
    [ID_TIPOIE] [int] NOT NULL,
```

```
    [TIE_DESCRIPCION] [varchar](50) NULL,
```

```
    [TIE_ESTADO] [varchar](50) NULL,
```

CONSTRAINT [PK\_TBL\_TIPO\_IE] PRIMARY KEY CLUSTERED

(

[ID\_TIPOIE] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_TIPO\_EMPLEADO] Script Date: 10/11/2014

18:55:29 \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_TIPO\_EMPLEADO](



```
[ID_TIPOEMPLEADO] [int] NOT NULL,  
  
[TEM_TIPO] [varchar](50) NULL,  
  
[TEM_CARGO] [varchar](50) NULL,  
  
[TEM_ESTADO] [varchar](50) NULL,  
  
CONSTRAINT [PK_TBL_TIPO_EMPLEADO] PRIMARY KEY CLUSTERED  
  
(  
  
    [ID_TIPOEMPLEADO] ASC  
  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
  
) ON [PRIMARY]  
  
GO  
  
SET ANSI_PADDING OFF  
  
GO  
  
/***** Object: Table [dbo].[TBL_TIPO_DOCUMENTOS]   Script Date:  
10/11/2014 18:55:29 *****/  
  
SET ANSI_NULLS ON  
  
GO  
  
SET QUOTED_IDENTIFIER ON
```

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_TIPO\_DOCUMENTOS](

[ID\_TIPODOCUMENTOS] [int] NOT NULL,

[TDO\_DESCRIPCION] [varchar](50) NULL,

[TDO\_ESTADO] [varchar](50) NULL,

CONSTRAINT [PK\_TBL\_TIPO\_DOCUMENTOS] PRIMARY KEY

CLUSTERED

(

[ID\_TIPODOCUMENTOS] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_RUBROS] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_RUBROS](

[ID\_RUBROS] [int] NOT NULL,

[RUB\_DETALLE] [varchar](50) NULL,

[RUB\_NOMBRE] [varchar](50) NULL,

[RUB\_VALOR] [decimal](18, 0) NULL,

[RUB\_INGRESO] [decimal](18, 0) NULL,

[RUB\_EGRESO] [decimal](18, 0) NULL,

[RUB\_ESTADO] [varchar](50) NULL,

[RUB\_RECIB] [decimal](18, 0) NULL,

CONSTRAINT [PK\_TBL\_RUBROS] PRIMARY KEY CLUSTERED

(

[ID\_RUBROS] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[TBL_PERFIL] Script Date: 10/11/2014 18:55:29
```

```
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_PERFIL](
```

```
    [ID_PERFIL] [int] NOT NULL,
```

```
    [PER_NOMBRE] [varchar](50) NULL,
```

```
    [PER_FECHA] [date] NULL,
```

```
[PER_ESTADO] [varchar](50) NULL,  
  
CONSTRAINT [PK_TBL_PERFIL] PRIMARY KEY CLUSTERED  
  
(  
  
    [ID_PERFIL] ASC  
  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
  
) ON [PRIMARY]  
  
GO  
  
SET ANSI_PADDING OFF  
  
GO  
  
/***** Object: Table [dbo].[TBL_AUDITORIA]   Script Date: 10/11/2014  
18:55:29 *****/  
  
SET ANSI_NULLS ON  
  
GO  
  
SET QUOTED_IDENTIFIER ON  
  
GO  
  
SET ANSI_PADDING ON  
  
GO
```



```
CREATE TABLE [dbo].[TBL_AUDITORIA](  
  
    [iD_AUDITORIA] [int] NOT NULL,  
  
    [AUD_USER] [varchar](50) NULL,  
  
    [AUD_MODALO] [varchar](50) NULL,  
  
    [AUD_OBJ] [varchar](50) NULL,  
  
    [AUD_AUD_FECHA] [varchar](50) NULL,  
  
    [AUD_STATION] [varchar](50) NULL,  
  
    [AUD_ESTADO] [varchar](50) NULL,  
  
    CONSTRAINT [PK_TBL_AUDITORIA] PRIMARY KEY CLUSTERED  
  
    (  
  
        [iD_AUDITORIA] ASC  
  
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
    ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
  
    ) ON [PRIMARY]  
  
GO  
  
SET ANSI_PADDING OFF  
  
GO
```

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_CIUADAD] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_CIUADAD](

[ID\_CIUADAD] [int] NOT NULL,

[CIU\_DESCRIPCION] [varchar](50) NULL,

[CIU\_ESTADO] [varchar](50) NULL,

CONSTRAINT [PK\_TBL\_CIUADAD] PRIMARY KEY CLUSTERED

(

[ID\_CIUADAD] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_BODEGA] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_BODEGA](

[ID\_BODEGA] [int] NOT NULL,

[BOD\_DESCRIPCION] [varchar](50) NULL,

[BOD\_ESTADO] [varchar](50) NULL,

[ID\_CIUADAD] [int] NULL,

CONSTRAINT [PK\_TBL\_BODEGA] PRIMARY KEY CLUSTERED

(

[ID\_BODEGA] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[TBL_EMPLEADO] Script Date: 10/11/2014
```

```
18:55:29 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_EMPLEADO](
```

```
    [ID_EMPLEADO] [int] NOT NULL,
```

```
    [EMP_NOMBRES] [varchar](50) NULL,
```

```
    [EMP_APELLIDOS] [varchar](50) NULL,
```



[EMP\_DIRECCION] [varchar](50) NULL,

[EMP\_TELEFONO] [varchar](50) NULL,

[EMP\_FECHNAC] [date] NULL,

[EMP\_CEDULA] [varchar](50) NULL,

[EMP\_CORREO] [varchar](50) NULL,

[EMP\_IMAGEN] [varchar](50) NULL,

[EMP\_ESTADO] [varchar](50) NULL,

[ID\_TIPOEMPLEADO] [int] NULL,

[ID\_CIUDAD] [int] NULL,

CONSTRAINT [PK\_TBL\_EMPLEADO] PRIMARY KEY CLUSTERED

(

[ID\_EMPLEADO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_DOCUMENTOS] Script Date: 10/11/2014

18:55:29 \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_DOCUMENTOS](

[ID\_DOCUMENTOS] [int] NOT NULL,

[DOC\_DESCRIPCION] [varchar](50) NULL,

[DOC\_FECHA] [date] NULL,

[DOC\_ESTADO] [varchar](50) NULL,

[ID\_TIPODOCUMENTOS] [int] NULL,

CONSTRAINT [PK\_TBL\_DOCUMENTOS] PRIMARY KEY CLUSTERED

(

[ID\_DOCUMENTOS] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[TBL_DETALLEROL] Script Date: 10/11/2014
```

```
18:55:29 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_DETALLEROL](
```

```
    [ID_DETALLEROL] [int] NOT NULL,
```

```
    [DRO_TOTALINGRESOS] [decimal](18, 0) NULL,
```

```
    [DRO_TOTALEGRESOS] [decimal](18, 0) NULL,
```

[DRO\_MONTO] [decimal](18, 0) NULL,

[ID\_RUBROS] [int] NULL,

[DRO\_ESTADO] [varchar](50) NULL,

CONSTRAINT [PK\_TBL\_DETALLEROL] PRIMARY KEY CLUSTERED

(

[ID\_DETALLEROL] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_USUARIO] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_USUARIO](
```

```
    [ID_USUARIO] [int] NOT NULL,
```

```
    [USU_NOMBRE] [varchar](50) NULL,
```

```
    [USU_CLAVE] [varchar](50) NULL,
```

```
    [USU_EMAIL] [varchar](50) NULL,
```

```
    [USU_FECHACREACION] [date] NULL,
```

```
    [USU_ESTADO] [varchar](50) NULL,
```

```
    [ID_PERFIL] [int] NULL,
```

```
    CONSTRAINT [PK_TBL_USUARIO] PRIMARY KEY CLUSTERED
```

```
(
```

```
    [ID_USUARIO] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
```

```
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
```

```
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```



GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_ROL] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_ROL](

[ID\_ROL] [int] NOT NULL,

[ROL\_FECHACREACION] [date] NULL,

[ROL\_ESTADO] [varchar](50) NULL,

[ID\_RUBROS] [int] NULL,

[ID\_EMPLEADO] [int] NULL,

CONSTRAINT [PK\_TBL\_ROL] PRIMARY KEY CLUSTERED

(

[ID\_ROL] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[TBL_RECURSOS] Script Date: 10/11/2014
```

```
18:55:29 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[TBL_RECURSOS](
```

```
    [ID_RECURSOS] [int] NOT NULL,
```

```
    [REC_DESCRIPCION] [varchar](50) NULL,
```

```
    [REC_ESTADO] [varchar](50) NULL,
```

[REC\_FECHACREACION] [date] NULL,

[REC\_STOCK] [int] NULL,

[ID\_TIPORECURSOS] [int] NULL,

[ID\_BODEGA] [int] NULL,

CONSTRAINT [PK\_TBL\_RECURSOS] PRIMARY KEY CLUSTERED

(

[ID\_RECURSOS] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_INFORMES] Script Date: 10/11/2014

18:55:29 \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_INFORMES](

[ID\_INFORMES] [int] NOT NULL,

[INF\_DESCRIPCION] [varchar](50) NULL,

[INF\_FECHA] [date] NULL,

[INF\_ESTADO] [varchar](50) NULL,

[ID\_BODEGA] [int] NULL,

[ID\_DOCUMENTOS] [int] NULL,

[ID\_EMPLEADO] [int] NULL,

CONSTRAINT [PK\_TBL\_INFORMES] PRIMARY KEY CLUSTERED

(

[ID\_INFORMES] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_INGRESO\_EGRESO] Script Date:

10/11/2014 18:55:29 \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_INGRESO\_EGRESO](

[ID\_INGRESOEGRESO] [int] NOT NULL,

[INE\_DESCRIPCION] [varchar](50) NULL,

[INE\_FECHA] [date] NULL,

[INE\_CANTIDAD] [int] NULL,

[INE\_ESTADO] [varchar](50) NULL,

[ID\_TIPOIE] [int] NULL,

[ID\_RECURSOS] [int] NULL,

CONSTRAINT [PK\_TBL\_INGRESO\_EGRESO] PRIMARY KEY CLUSTERED

```
(  
  
    [ID_INGRESOEGRESO] ASC  
  
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
    ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
  
    ) ON [PRIMARY]  
  
GO  
  
SET ANSI_PADDING OFF  
  
GO  
  
/***** Object: Table [dbo].[TBL_DETALLE]   Script Date: 10/11/2014 18:55:29  
*****/  
  
SET ANSI_NULLS ON  
  
GO  
  
SET QUOTED_IDENTIFIER ON  
  
GO  
  
SET ANSI_PADDING ON  
  
GO  
  
CREATE TABLE [dbo].[TBL_DETALLE](  
  
    [ID_DETALLE] [int] NOT NULL,
```

[DET\_CLAVE] [varchar](50) NULL,  
  
[DET\_FECHA] [date] NULL,  
  
[DET\_DETALLE] [varchar](50) NULL,  
  
[DET\_OBSERVACION] [varchar](50) NULL,  
  
[DET\_COSTOUNIT] [decimal](18, 0) NULL,  
  
[DET\_TOTAL] [int] NULL,  
  
[DET\_COSTOTOTAL] [decimal](18, 0) NULL,  
  
[DET\_IMAGEN] [varchar](50) NULL,  
  
[ID\_RECURSOS] [int] NULL,  
  
[DET\_ESTADO] [varchar](50) NULL,

CONSTRAINT [PK\_TBL\_DETALLE] PRIMARY KEY CLUSTERED

(

[ID\_DETALLE] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\* Object: Table [dbo].[TBL\_QR] Script Date: 10/11/2014 18:55:29

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[TBL\_QR](

[ID\_QR] [int] NOT NULL,

[QR\_DESCRIPCION] [varchar](50) NULL,

[QR\_IMAGEN] [varchar](50) NULL,

[QR\_ESTADO] [varchar](50) NULL,

[ID\_RECURSOS] [int] NULL,

CONSTRAINT [PK\_TBL\_QR] PRIMARY KEY CLUSTERED

(

[ID\_QR] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_BODEGA_TBL_CIUADAD] Script Date:  
10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_BODEGA] WITH CHECK ADD CONSTRAINT  
[FK_TBL_BODEGA_TBL_CIUADAD] FOREIGN KEY([ID_CIUADAD])  
REFERENCES [dbo].[TBL_CIUADAD] ([ID_CIUADAD])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_BODEGA] CHECK CONSTRAINT  
[FK_TBL_BODEGA_TBL_CIUADAD]
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_DETALLE_TBL_RECURSOS] Script  
Date: 10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_DETALLE] WITH CHECK ADD CONSTRAINT  
[FK_TBL_DETALLE_TBL_RECURSOS] FOREIGN KEY([ID_RECURSOS])  
REFERENCES [dbo].[TBL_RECURSOS] ([ID_RECURSOS])
```

GO

```
ALTER TABLE [dbo].[TBL_DETALLE] CHECK CONSTRAINT  
[FK_TBL_DETALLE_TBL_RECURSOS]
```

GO

```
/****** Object: ForeignKey [FK_TBL_DETALLEROL_TBL_RUBROS] Script
```

```
Date: 10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_DETALLEROL] WITH CHECK ADD  
CONSTRAINT [FK_TBL_DETALLEROL_TBL_RUBROS] FOREIGN  
KEY([ID_RUBROS])
```

```
REFERENCES [dbo].[TBL_RUBROS] ([ID_RUBROS])
```

GO

```
ALTER TABLE [dbo].[TBL_DETALLEROL] CHECK CONSTRAINT  
[FK_TBL_DETALLEROL_TBL_RUBROS]
```

GO

```
/****** Object: ForeignKey
```

```
[FK_TBL_DOCUMENTOS_TBL_TIPO_DOCUMENTOS] Script Date:
```

```
10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_DOCUMENTOS] WITH CHECK ADD  
CONSTRAINT [FK_TBL_DOCUMENTOS_TBL_TIPO_DOCUMENTOS]  
FOREIGN KEY([ID_TIPODOCUMENTOS])
```

```
REFERENCES [dbo].[TBL_TIPO_DOCUMENTOS] ([ID_TIPODOCUMENTOS])
```

GO

```
ALTER TABLE [dbo].[TBL_DOCUMENTOS] CHECK CONSTRAINT  
[FK_TBL_DOCUMENTOS_TBL_TIPO_DOCUMENTOS]
```

GO

/\*\*\*\*\* Object: ForeignKey [FK\_TBL\_EMPLEADO\_TBL\_CIUADAD] Script

Date: 10/11/2014 18:55:29 \*\*\*\*\*/

```
ALTER TABLE [dbo].[TBL_EMPLEADO] WITH CHECK ADD CONSTRAINT  
[FK_TBL_EMPLEADO_TBL_CIUADAD] FOREIGN KEY([ID_CIUADAD])
```

```
REFERENCES [dbo].[TBL_CIUADAD] ([ID_CIUADAD])
```

GO

```
ALTER TABLE [dbo].[TBL_EMPLEADO] CHECK CONSTRAINT  
[FK_TBL_EMPLEADO_TBL_CIUADAD]
```

GO

/\*\*\*\*\* Object: ForeignKey [FK\_TBL\_EMPLEADO\_TBL\_TIPO\_EMPLEADO]

Script Date: 10/11/2014 18:55:29 \*\*\*\*\*/

```
ALTER TABLE [dbo].[TBL_EMPLEADO] WITH CHECK ADD CONSTRAINT  
[FK_TBL_EMPLEADO_TBL_TIPO_EMPLEADO] FOREIGN
```

```
KEY([ID_TIPOEMPLEADO])
```

```
REFERENCES [dbo].[TBL_TIPO_EMPLEADO] ([ID_TIPOEMPLEADO])
```

GO

```
ALTER TABLE [dbo].[TBL_EMPLEADO] CHECK CONSTRAINT  
[FK_TBL_EMPLEADO_TBL_TIPO_EMPLEADO]
```

GO

```
/****** Object: ForeignKey [FK_TBL_INFORMES_TBL_BODEGA] Script  
Date: 10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_INFORMES] WITH CHECK ADD CONSTRAINT  
[FK_TBL_INFORMES_TBL_BODEGA] FOREIGN KEY([ID_BODEGA])
```

```
REFERENCES [dbo].[TBL_BODEGA] ([ID_BODEGA])
```

GO

```
ALTER TABLE [dbo].[TBL_INFORMES] CHECK CONSTRAINT  
[FK_TBL_INFORMES_TBL_BODEGA]
```

GO

```
/****** Object: ForeignKey [FK_TBL_INFORMES_TBL_DOCUMENTOS]  
Script Date: 10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_INFORMES] WITH CHECK ADD CONSTRAINT  
[FK_TBL_INFORMES_TBL_DOCUMENTOS] FOREIGN
```

```
KEY([ID_DOCUMENTOS])
```

```
REFERENCES [dbo].[TBL_DOCUMENTOS] ([ID_DOCUMENTOS])
```

GO

```
ALTER TABLE [dbo].[TBL_INFORMES] CHECK CONSTRAINT  
[FK_TBL_INFORMES_TBL_DOCUMENTOS]
```



GO

/\*\*\*\*\* Object: ForeignKey [FK\_TBL\_INFORMES\_TBL\_EMPLEADO] Script

Date: 10/11/2014 18:55:29 \*\*\*\*\*/

ALTER TABLE [dbo].[TBL\_INFORMES] WITH CHECK ADD CONSTRAINT  
[FK\_TBL\_INFORMES\_TBL\_EMPLEADO] FOREIGN KEY([ID\_EMPLEADO])

REFERENCES [dbo].[TBL\_EMPLEADO] ([ID\_EMPLEADO])

GO

ALTER TABLE [dbo].[TBL\_INFORMES] CHECK CONSTRAINT  
[FK\_TBL\_INFORMES\_TBL\_EMPLEADO]

GO

/\*\*\*\*\* Object: ForeignKey [FK\_TBL\_INGRESO\_EGRESO\_TBL\_RECURSOS]

Script Date: 10/11/2014 18:55:29 \*\*\*\*\*/

ALTER TABLE [dbo].[TBL\_INGRESO\_EGRESO] WITH CHECK ADD  
CONSTRAINT [FK\_TBL\_INGRESO\_EGRESO\_TBL\_RECURSOS] FOREIGN  
KEY([ID\_RECURSOS])

REFERENCES [dbo].[TBL\_RECURSOS] ([ID\_RECURSOS])

GO

ALTER TABLE [dbo].[TBL\_INGRESO\_EGRESO] CHECK CONSTRAINT  
[FK\_TBL\_INGRESO\_EGRESO\_TBL\_RECURSOS]

GO

/\*\*\*\*\*\* Object: ForeignKey [FK\_TBL\_INGRESO\_EGRESO\_TBL\_TIPO\_IE]

Script Date: 10/11/2014 18:55:29 \*\*\*\*\*/

```
ALTER TABLE [dbo].[TBL_INGRESO_EGRESO] WITH CHECK ADD  
CONSTRAINT [FK_TBL_INGRESO_EGRESO_TBL_TIPO_IE] FOREIGN  
KEY([ID_TIPOIE])
```

```
REFERENCES [dbo].[TBL_TIPO_IE] ([ID_TIPOIE])
```

GO

```
ALTER TABLE [dbo].[TBL_INGRESO_EGRESO] CHECK CONSTRAINT  
[FK_TBL_INGRESO_EGRESO_TBL_TIPO_IE]
```

GO

/\*\*\*\*\*\* Object: ForeignKey [FK\_TBL\_QR\_TBL\_RECURSOS] Script Date:

10/11/2014 18:55:29 \*\*\*\*\*/

```
ALTER TABLE [dbo].[TBL_QR] WITH CHECK ADD CONSTRAINT  
[FK_TBL_QR_TBL_RECURSOS] FOREIGN KEY([ID_RECURSOS])
```

```
REFERENCES [dbo].[TBL_RECURSOS] ([ID_RECURSOS])
```

GO

```
ALTER TABLE [dbo].[TBL_QR] CHECK CONSTRAINT  
[FK_TBL_QR_TBL_RECURSOS]
```

GO

/\*\*\*\*\*\* Object: ForeignKey [FK\_TBL\_RECURSOS\_TBL\_BODEGA] Script

Date: 10/11/2014 18:55:29 \*\*\*\*\*/

```
ALTER TABLE [dbo].[TBL_RECURSOS] WITH CHECK ADD CONSTRAINT  
[FK_TBL_RECURSOS_TBL_BODEGA] FOREIGN KEY([ID_BODEGA])
```

```
REFERENCES [dbo].[TBL_BODEGA] ([ID_BODEGA])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_RECURSOS] CHECK CONSTRAINT
```

```
[FK_TBL_RECURSOS_TBL_BODEGA]
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_RECURSOS_TBL_TIPO_RECURSOS]
```

```
Script Date: 10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_RECURSOS] WITH CHECK ADD CONSTRAINT  
[FK_TBL_RECURSOS_TBL_TIPO_RECURSOS] FOREIGN
```

```
KEY([ID_TIPORECURSOS])
```

```
REFERENCES [dbo].[TBL_TIPO_RECURSOS] ([ID_TIPORECURSOS])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_RECURSOS] CHECK CONSTRAINT
```

```
[FK_TBL_RECURSOS_TBL_TIPO_RECURSOS]
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_ROL_TBL_EMPLEADO] Script Date:
```

```
10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_ROL] WITH CHECK ADD CONSTRAINT
```

```
[FK_TBL_ROL_TBL_EMPLEADO] FOREIGN KEY([ID_EMPLEADO])
```

```
REFERENCES [dbo].[TBL_EMPLEADO] ([ID_EMPLEADO])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_ROL] CHECK CONSTRAINT
```

```
[FK_TBL_ROL_TBL_EMPLEADO]
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_ROL_TBL_RUBROS] Script Date:
```

```
10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_ROL] WITH CHECK ADD CONSTRAINT
```

```
[FK_TBL_ROL_TBL_RUBROS] FOREIGN KEY([ID_RUBROS])
```

```
REFERENCES [dbo].[TBL_RUBROS] ([ID_RUBROS])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_ROL] CHECK CONSTRAINT
```

```
[FK_TBL_ROL_TBL_RUBROS]
```

```
GO
```

```
/****** Object: ForeignKey [FK_TBL_USUARIO_TBL_PERFIL] Script Date:
```

```
10/11/2014 18:55:29 *****/
```

```
ALTER TABLE [dbo].[TBL_USUARIO] WITH CHECK ADD CONSTRAINT
```

```
[FK_TBL_USUARIO_TBL_PERFIL] FOREIGN KEY([ID_PERFIL])
```

```
REFERENCES [dbo].[TBL_PERFIL] ([ID_PERFIL])
```

```
GO
```

```
ALTER TABLE [dbo].[TBL_USUARIO] CHECK CONSTRAINT  
[FK_TBL_USUARIO_TBL_PERFIL]
```

```
GO
```



INSTITUTO TECNOLÓGICO  
"CORDILLERA"

CARRERA ANÁLISIS DE SISTEMAS

CONTROL DE PROCESOS DE INVENTARIO Y NÓMINA MEDIANTE UN  
SISTEMA WEB PARA LA EMPRESA CONSTRUCTUM S. A UBICADA EN LA  
CIUDAD DE QUITO

Manual de usuario

Autor: Peñafiel Asimbaya Bryan Patricio

Tutor: Ing. Hugo Heredia

Quito, Octubre 2014

## ÍNDICE

Objetivo del manual .....	101
Introducción .....	102
Ejecución de la aplicación.....	102
Submenú T.T.H.H.....	104
Personal .....	104
Administración.....	107
Documentos .....	108
Bodega.....	110



## ÍNDICE DE FIGURAS

Figura 1: Ventana de acceso al sistema .....	102
Figura 2: Menú del sistema.....	103
Figura 3: Submenú de T.T.H.H.....	104
Figura 4: interfaz personal .....	104
Figura 5: Interfaz personal .....	105
Figura 6: interfaz T.T.H.H. ....	105
Figura 7: Interfaz rol de pagos .....	106
Figura 8: Interfaz administración.....	107
Figura 9: Ventana de registro de usuarios.....	108
Figura 10: Interfaz documentos .....	108
Figura 11: Ventana de documentos.....	109
Figura 12: Interfaz bodega .....	110
Figura 13: Interfaz de bodega .....	110
Figura 14: Interfaz material herramienta .....	111
Figura 15: Códigos Qr .....	112
Figura 16: Interfaz recursos de bodega.....	112
Figura 17: Carpeta códigos Qr.....	113
Figura 18: Interfaz control de entradas y salidas. ....	114
Figura 19: Interfaz control de recursos .....	114

## MANUAL SISTEMA DE CONTROL

### Objetivo del manual

El objetivo de la elaboración de este manual es guiar al usuario a utilizar el sistema de control de inventario por medio de información útil que el usuario necesita para lograr obtener buenos resultados con el sistema mediante los siguientes puntos:

- Introducción
- Requerimientos
- Ejecución de la aplicación
- Manipulación del sistema
- Simulación del seguimiento de una trayectoria Almacenada

### Dirigido a

Este manual está orientado a una parte de los Usuarios, trabajadores internos del negocio.

Los conocimientos mínimos que deben tener las personas que operan los formularios y deberán utilizar este manual son:

- Conocimientos básicos de un sistema de inventario.
- Conocimientos básicos de Windows.

### Generalidades

La interfaz de visualización del sistema de control se implementó con fines de mejora y avance en donde se visualiza el seguimiento de una trayectoria de datos, empleando para ello el control de sistema y una mejor perspectiva de llevar un control con facilidad. Estas interfaces se desarrollaron empleando páginas web en formato aspx y programado en Visual Estudio 2012.

## Introducción

El objetivo de este manual es dar a conocer cuáles son las estrategias y la forma de interactuar datos al cabo de tener un control de estos ya ingresados, además de conocer y apreciar la facilidad de un sistema de control.

## Requerimientos

Para poder hacer un buen uso del sistema se tiene que instalar lo siguiente:

- Sistema Operativo – Windows Seven.
- Framework – versión superior a 4.0 o superior
- Sql server 2008.

## Ejecución de la aplicación

La aplicación se ejecuta al ingresar a la siguiente dirección <http://localhost:6077/Login.aspx> desde cualquier navegador de su preferencia, al ingresar a la aplicación, se abre una ventana la cual será para el ingreso al sistema mediante un nombre y una clave la cual se tendrá que tipiar y realizar un enter en cada caja de texto para saltar a la siguiente, una vez ingresado los dos campos se puede realizar un click en el botón o un enter.

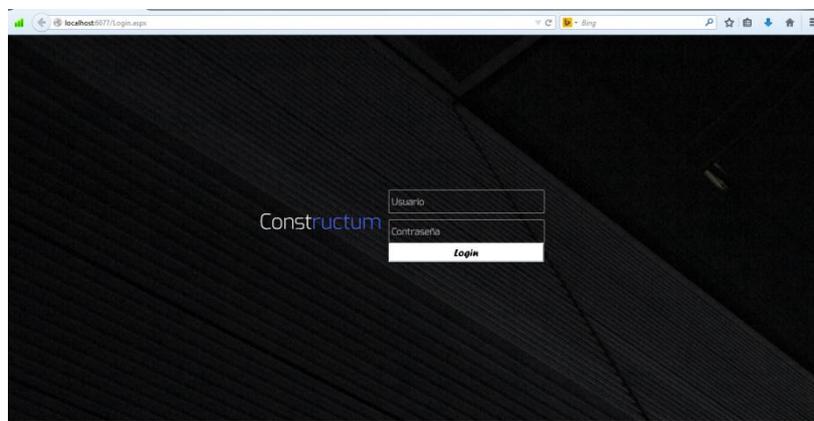


Figura 1: Ventana de acceso al sistema

Después de la ventana de ingreso nos mostrará la ventana del menú principal la cual nos mostrará las opciones que tendrá el sistema a realizar.



Figura 2: Menú del sistema

En esta ventana se puede observar las opciones:

- A) Administración (El mantenimiento de cada uno de los usuarios que podrán trabajar con el sistema).
- B) Documentos (Ingreso, modificación y eliminación de documentos del negocio).
- C) Personal (se ejecuta el pago de personal, así como el mantenimiento de los trabajadores del negocio).
- D) Bodega (Se realiza el mantenimiento de material al igual que el de herramienta los cuales se podrán ingresar al sistema)

Para acceder a la interfaz de cada uno de ellos solo es necesario dar un click encima de cada link o botón que se presenta en pantalla.

Al realizar un click en personal o bodega nos desplegará un submenú el cual nos mostrará las opciones que cada interfaz presenta o un cuadro con los datos de cada interfaz.

## Submenú T.T.H.H

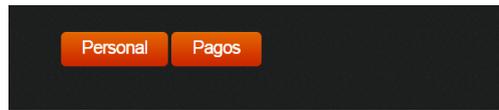


Figura 3: Submenú de T.T.H.H

Al realizar click en personal mostrará la interfaz de personal.

Al realizar click en pagos mostrará la interfaz de pagos.

## Personal

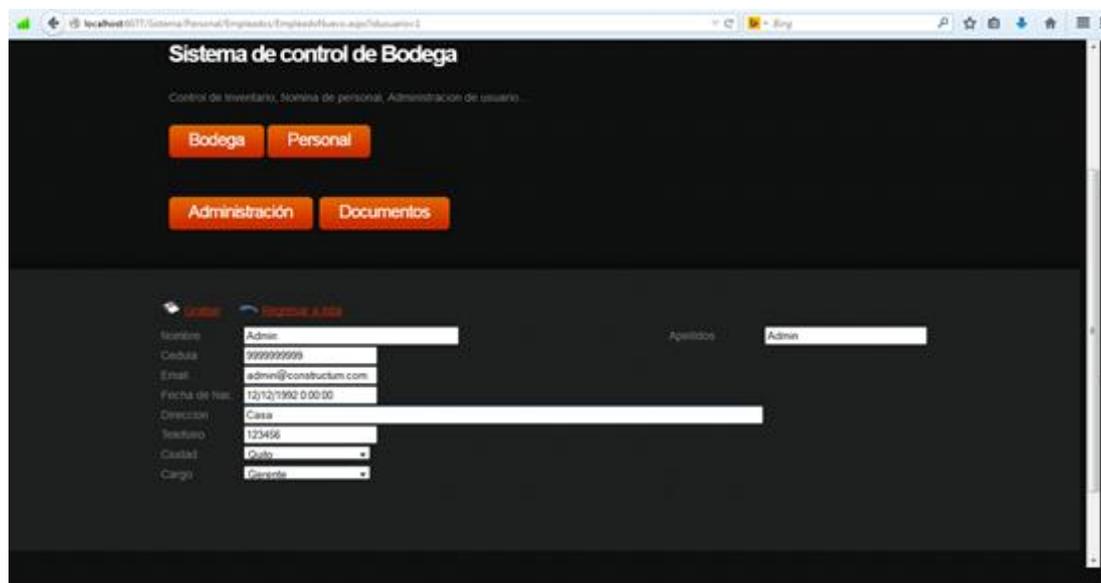


Figura 4: interfaz personal

Al realizar click en nuevo se desplegará otra pantalla para poder ingresar los datos del trabajador.

Al escoger un criterio en el combo box podemos buscar un trabajador ya sea por apellido o código.

Al realizar click en buscar nos mostrara el resultado.

En cada uno de los trabajadores podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que permite eliminar al trabajador.

Pantalla para ingresar nuevo trabajador o editar sus datos

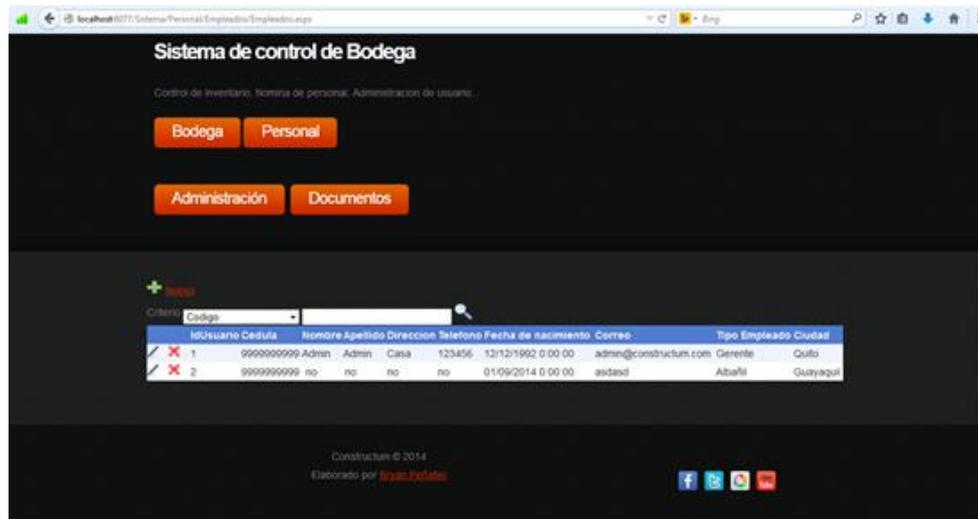


Figura 5: Interfaz personal

En esta pantalla observaremos los datos a editar del trabajador o simplemente un ingreso de datos para un nuevo trabajador.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo trabajador o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que ara el sistema es regresarnos al menú de personal.

### Pagos

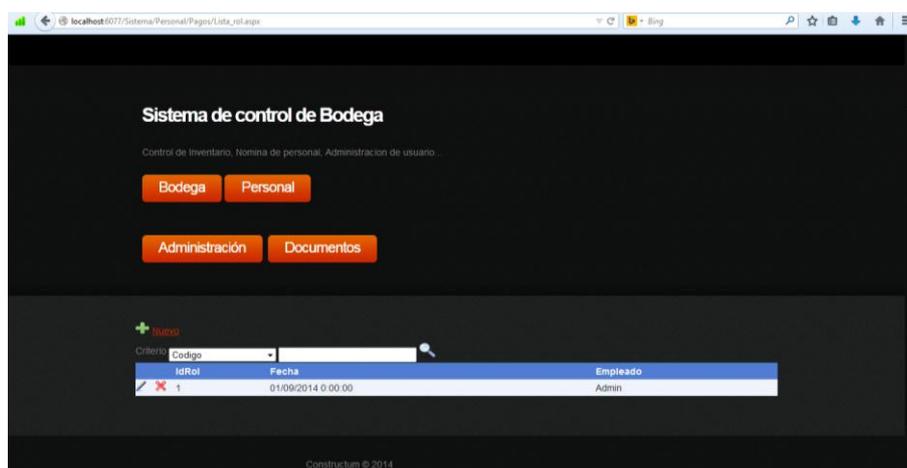


Figura 6: interfaz T.T.H.H.

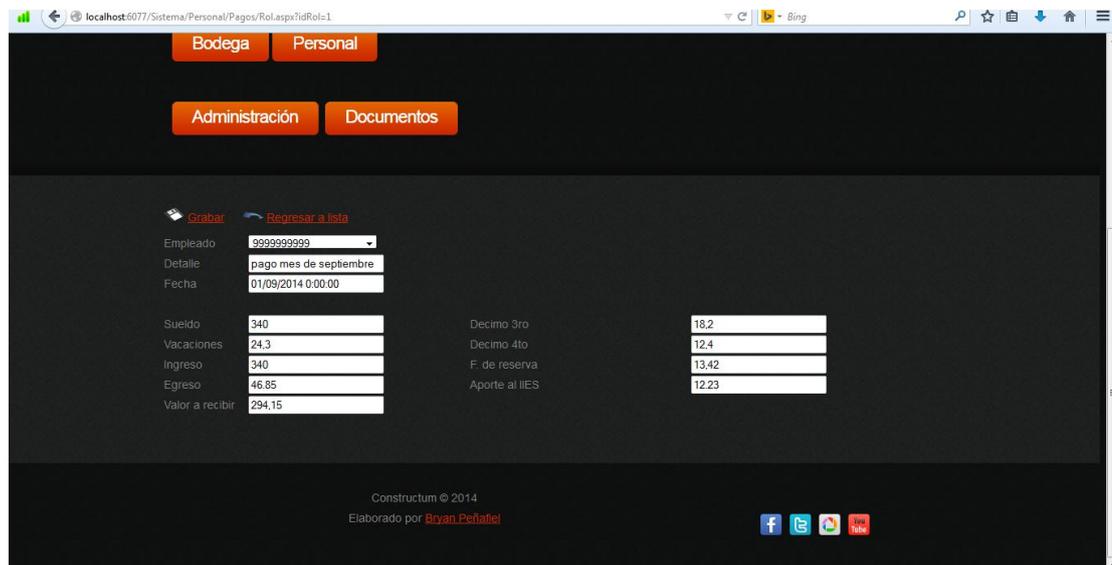
Al realizar click en nuevo nos desplegara otra pantalla para poder ingresar los datos del rol de pago.

Al escoger un criterio en el combo box podemos buscar un rol ya sea por empleado o código.

Al realizar click en buscar nos mostrara el resultado.

En cada uno de los roles de pago podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que nos permite eliminar el rol de pagos.

### **Pantalla para ingresar nuevo rol de pagos o editar sus datos**



Empleado	999999999
Detalle	pago mes de septiembre
Fecha	01/09/2014 0:00:00
Sueldo	340
Vacaciones	24.3
Ingreso	340
Egreso	46.85
Valor a recibir	294.15
Decimo 3ro	18.2
Decimo 4to	12.4
F. de reserva	13.42
Aporte al IIES	12.23

*Figura 7: Interfaz rol de pagos*

En esta pantalla observaremos los datos a editar del rol de pagos o simplemente un ingreso de datos para un nuevo rol.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo rol o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que ara el sistema es regresarnos al menú de rol de pagos.

## Administración

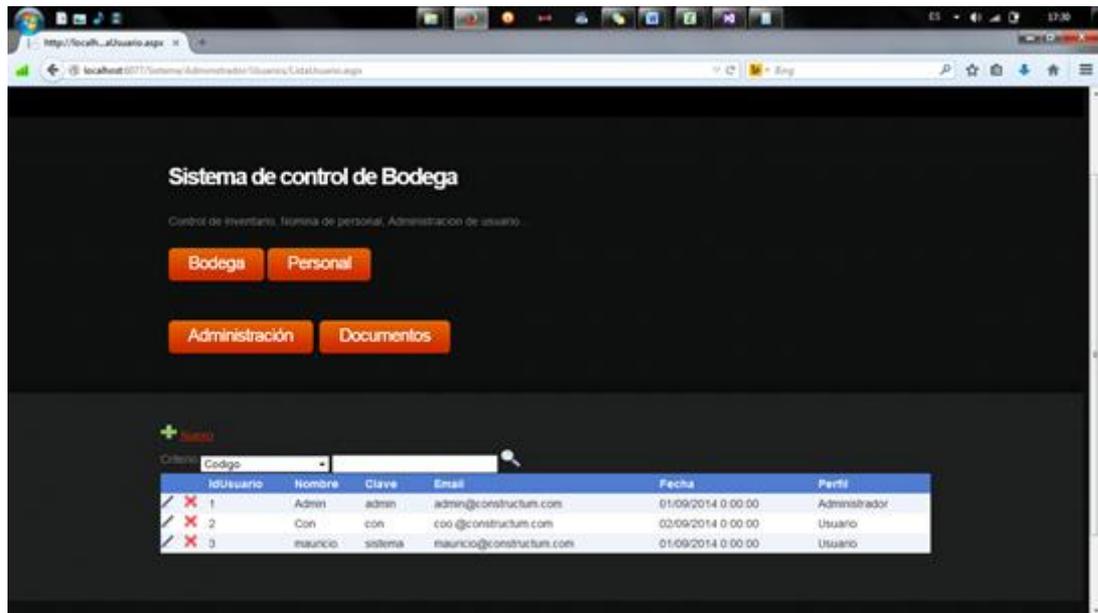


Figura 8: Interfaz administración

Al realizar click en nuevo nos desplegara otra pantalla para poder ingresar los datos del usuario.

Al escoger un criterio en el combo box podemos buscar un usuario ya sea por nombre o código.

Al realizar click en buscar nos mostrara el resultado.

En cada uno de los usuarios podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que nos permite eliminar al usuario.

## Pantalla para ingresar nuevo usuario o editar sus datos

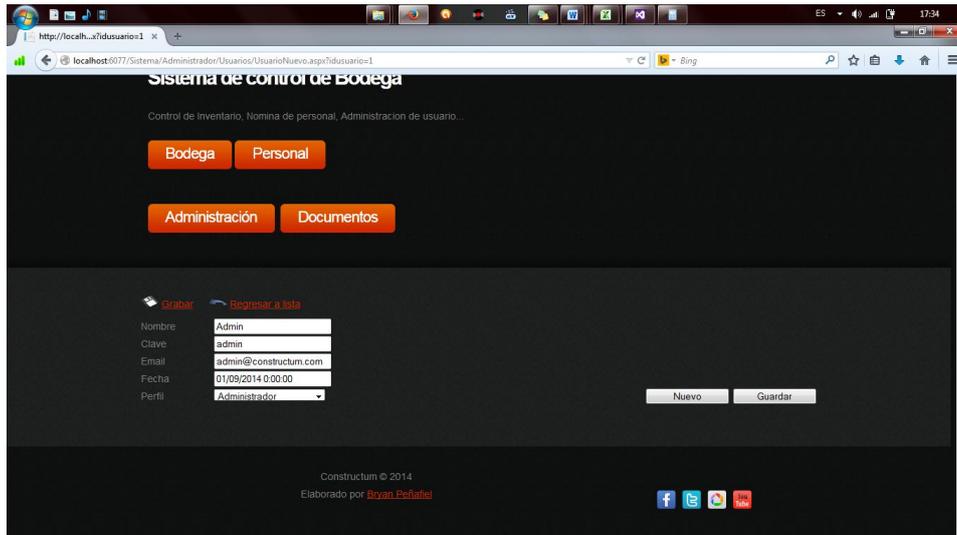


Figura 9: Ventana de registro de usuarios

En esta pantalla observaremos los datos a editar del usuario o simplemente un ingreso de datos para un nuevo usuario.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo usuario o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que hará el sistema es regresarnos al menú de Administración.

## Documentos

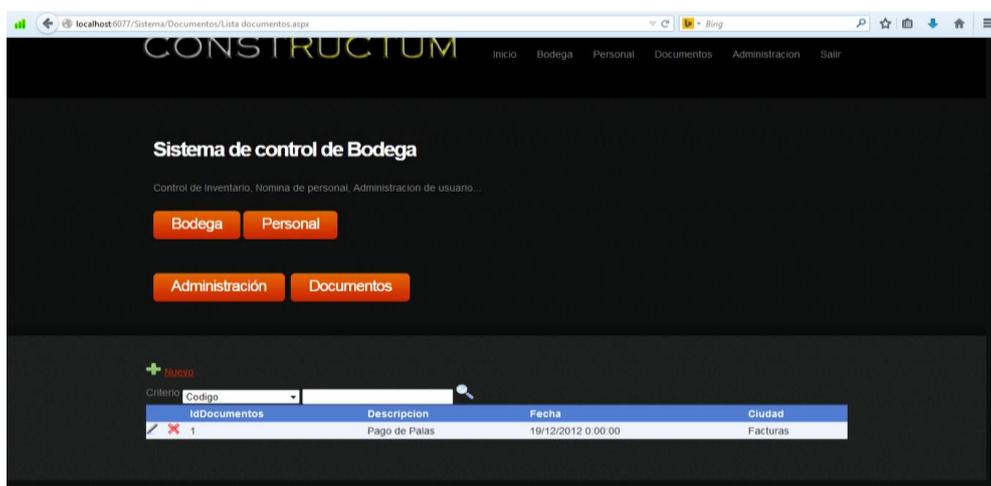


Figura 10: Interfaz documentos

Al realizar click en nuevo nos desplegara otra pantalla para poder ingresar los datos del documento.

Al escoger un criterio en el combo box podemos buscar un documento ya sea por descripción o código.

Al realizar click en buscar nos mostrara el resultado.

En cada uno de documentos podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que nos permite eliminar el documento.

### **Pantalla para ingresar nuevo documento o editar sus datos**



*Figura 11: Ventana de documentos*

En esta pantalla observaremos los datos a editar del documento o simplemente un ingreso de datos para un nuevo documento.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo documento o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que ara el sistema es regresarnos al menú de documentos.

## Bodega

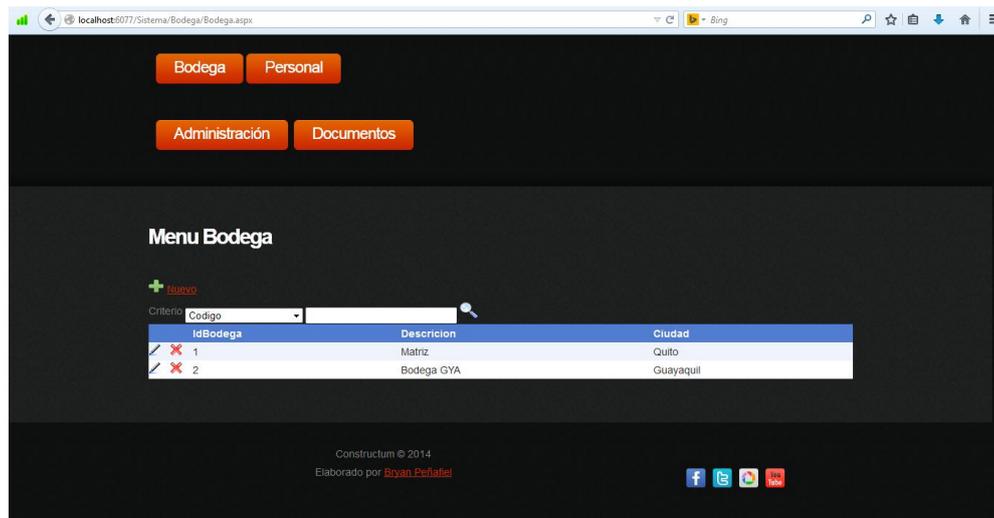


Figura 12: Interfaz bodega

Al realizar click en nuevo se desplegará otra pantalla para poder ingresar los datos de bodega.

Al escoger un criterio en el combo box podemos buscar una bodega ya sea por descripción o código.

Al realizar click en buscar mostrará el resultado.

En cada una de las bodegas podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que permite eliminar la bodega.

### Pantalla para ingresar nueva bodega o editar sus datos

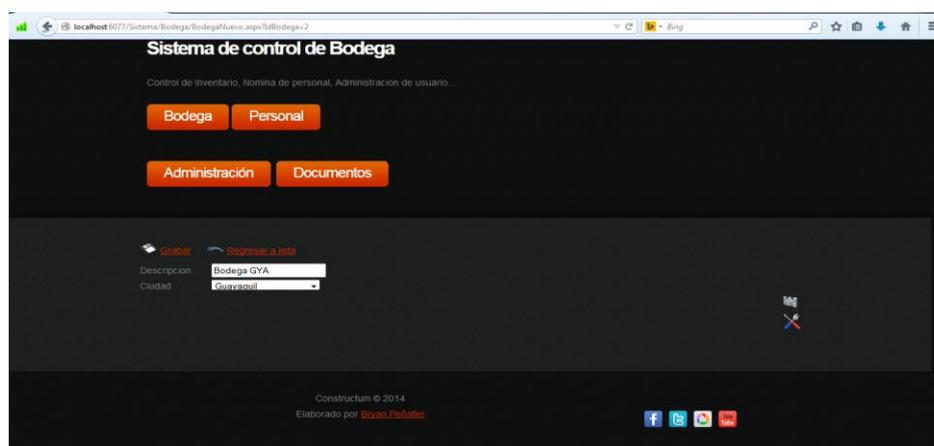


Figura 13: Interfaz de bodega

En esta pantalla observaremos los datos a editar de la bodega o simplemente un ingreso de datos para una nueva bodega.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificará si es un guardado de nuevo documento o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que hará el sistema es regresarnos al menú de bodega.

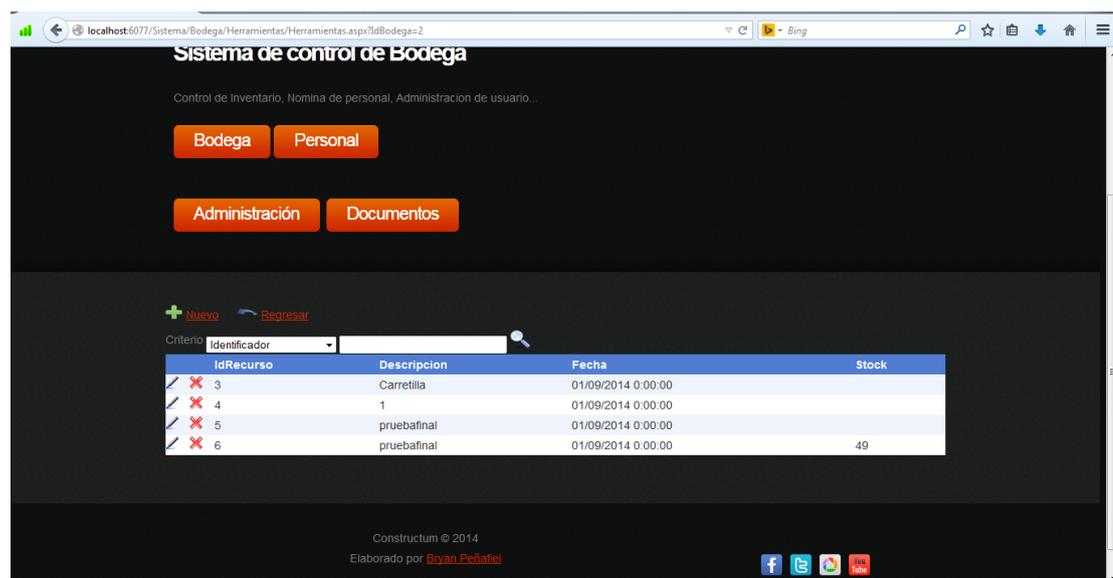
Como se puede observar en la parte inferior derecha un poco más arriba de las imágenes de redes sociales, se puede observar dos imágenes, las cuales son el material y herramienta que cada bodega contiene.

Al realizar click en la imagen de pared ingresaremos al menú material.

Al realizar click en la imagen de las herramientas ingresaremos al menú herramientas.

Cabe aclarar que los menús son exactamente iguales ya que el sistema lo que hace es un re direccionamiento ya sea para material o herramienta.

### Pantalla menú de material herramienta



The screenshot shows a web browser window with the URL `localhost:6077/Sistema/Bodega/Herramientas/Herramientas.aspx?idBodega=2`. The page title is "Sistema de control de Bodega" and the subtitle is "Control de Inventario, Nomina de personal, Administracion de usuario...". There are four orange buttons: "Bodega", "Personal", "Administración", and "Documentos". Below these are links for "+ Nuevo" and "Regresar". A search bar is present with the criterion "Identificador". A table displays the following data:

	IdRecurso	Descripcion	Fecha	Stock
✖	3	Carretilla	01/09/2014 0:00:00	
✖	4	1	01/09/2014 0:00:00	
✖	5	pruebafinai	01/09/2014 0:00:00	
✖	6	pruebafinai	01/09/2014 0:00:00	49

At the bottom, it says "Constructum © 2014" and "Elaborado por Bryan Peñafiel". There are also social media icons for Facebook, Twitter, YouTube, and Live.

Figura 14: Interfaz material herramienta

Al realizar click en nuevo se desplegará otra pantalla para poder ingresar los datos de material o herramienta.

Al escoger un criterio en el combo box podemos buscar un recurso ya sea por descripción o código.

Al realizar click en buscar mostrará un lector de códigos QR como el que se aprecia a continuación.

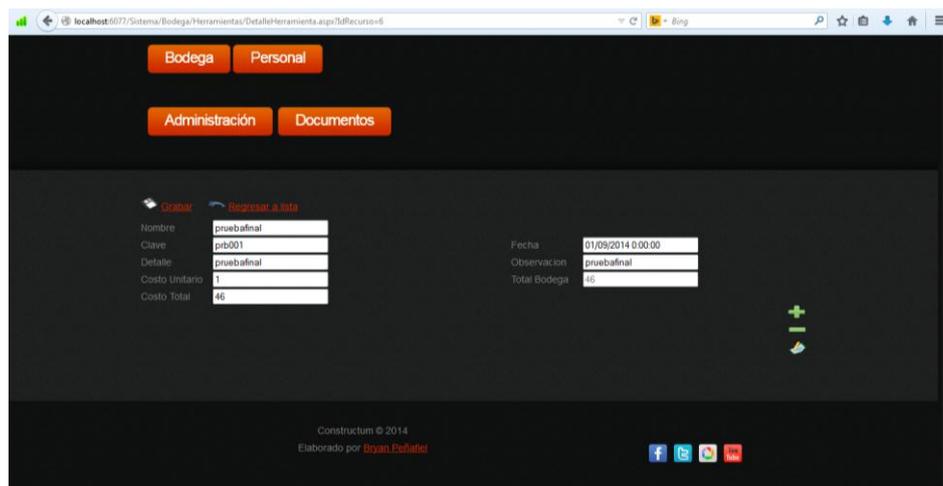


Figura 15: Códigos Qr

El cual automáticamente mostrará los resultados de la búsqueda.

En cada una de los recursos podemos observar que por su delante tiene un esfero que sirve para editar sus datos y una equis que permite eliminar los recursos.

### Pantalla para ingresar nuevo recurso o editar sus datos



Nombre	pruebafinal	Fecha	01/09/2014 0:00:00
Clave	prt001	Observacion	pruebafinal
Detalle	pruebafinal	Total Bodega	46
Costo Unitario	1		
Costo Total	46		

Figura 16: Interfaz recursos de bodega

En esta pantalla observaremos los datos a editar de los recursos o simplemente un ingreso de datos para un nuevo recurso.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificará si es un guardado de nuevo recurso o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que ara el sistema es regresarnos al menú del recurso en el que estemos trabajando.

Como se puede observar en la parte inferior derecha un poco más arriba de las imágenes de redes sociales, se puede observar tres imágenes, las cuales son el control de material y herramienta que cada bodega contiene.

Al realizar click en la imagen de más ingresaremos al menú entradas.

Al realizar click en la imagen de menos ingresaremos al menú salidas.

Al realizar click en la imagen de lápiz y papel ingresaremos al menú de historial o tipo kardex de las entradas y salidas del material.

Cabe aclarar que los menús Entradas y Salidas son exactamente iguales ya que el sistema lo que hace es un re direccionamiento ya sea por la una o por la otra.

Al guardar cada recurso automáticamente el sistema guardará una imagen del código QR, el cual lo identificará el sistema por el Id del recurso, en una carpeta especifica de Windows, como se muestra a continuación.



Figura 17: Carpeta códigos Qr

## Pantalla para ingresar nuevo Control (Entradas o Salidas) o editar sus datos

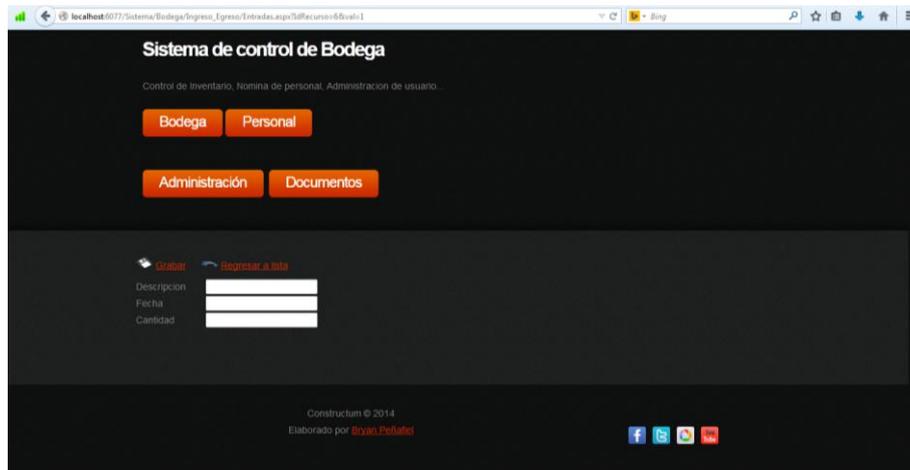


Figura 18: Interfaz control de entradas y salidas.

En esta pantalla observaremos los datos a editar del control o simplemente un ingreso de datos para un nuevo control.

Al realizar click en la imagen del disco duro o al link Grabar, el sistema identificara si es un guardado de nuevo control o una edición del mismo.

Al realizar click en la imagen de la flecha hacia tras o en el link regresar lista, lo que ara el sistema es regresarnos al menú del recurso en el que se esté trabajando.

## Pantalla de historial del control del recurso

	IdBodega	Descripcion	Fecha	Cantidad
✍ ✖	1	p	01/09/2014 0:00:00	4
✍ ✖	2	c	01/09/2014 0:00:00	2
✍ ✖	3	s	01/09/2014 0:00:00	3
✍ ✖	5	p	01/09/2014 0:00:00	6
✍ ✖	6	a	01/09/2014 0:00:00	1
✍ ✖	7	ff	01/09/2014 0:00:00	2
✍ ✖	8	q	01/09/2014 0:00:00	7

Figura 19: Interfaz control de recursos

Finalmente podemos apreciar el historial de control de entradas y salidas de cada uno de los recursos que estén registrados en el sistema.

En cada una de los ítems del historial de control podemos observar que por su delante tiene un esfero que sirve para editar sus datos en la pantalla re direccionada de ingreso o edición de control, y una equis que permite eliminar el ítem.



INSTITUTO TECNOLÓGICO  
"CORDILLERA"

CARRERA ANÁLISIS DE SISTEMAS

CONTROL DE PROCESOS DE INVENTARIO Y NÓMINA MEDIANTE UN  
SISTEMA WEB PARA LA EMPRESA CONSTRUCTUM S. A UBICADA EN LA  
CIUDAD DE QUITO

Manual técnico

Autor: Peñafiel Asimbaya Bryan Patricio

Tutor: Ing. Hugo Heredia

Quito, Octubre 2014



## ÍNDICE

Objetivo del Manual.....	118
1.1. Conexión .....	118
1.2 Capas del sistema .....	118
1.3 Lógica de negocios, Clase usuario .....	119
1.4 Pagina Login .....	122
1.5 Pagina para recursos.....	129
1.6 Lógica de negocios, Clase Ingreso_Egreso.....	134
1.7 Página historial.....	137
1.8 Script QR.....	141



## ÍNDICE DE FIGURAS

Figura 1: modelo de la base de datos .....	118
Figura 2: capas del sistema .....	118
Figura 3: Modelo de la base de datos .....	122
Figura 4: diseño de la página recursos .....	129
Figura 5: diseño de la página historial .....	137

## Objetivo del Manual

Proporcionar un resumen de la composición técnica de cada módulo y pantalla, así como los detalles técnicos de manera clara mediante la descripción y gráficos del sistema, para que los usuarios técnicos tengan una mayor visión de la estructuración y funcionamiento del sistema.

### 1.1. Conexión

La conexión del sistema se realizó mediante LINQ.

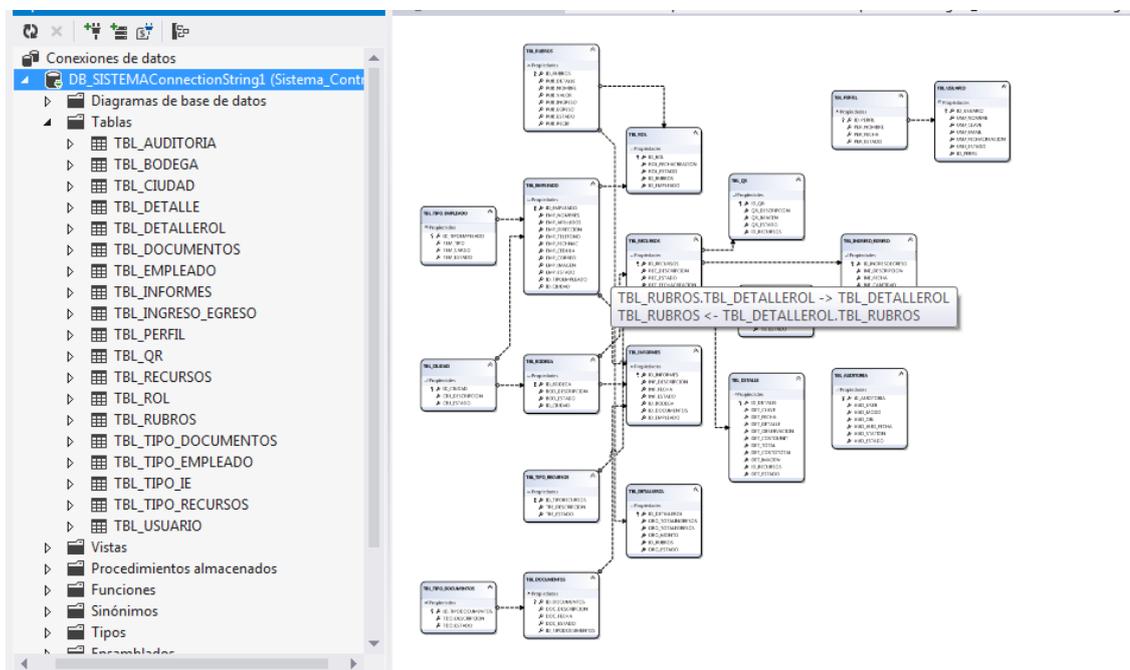


Figura 1: modelo de la base de datos

### 1.2 Capas del sistema

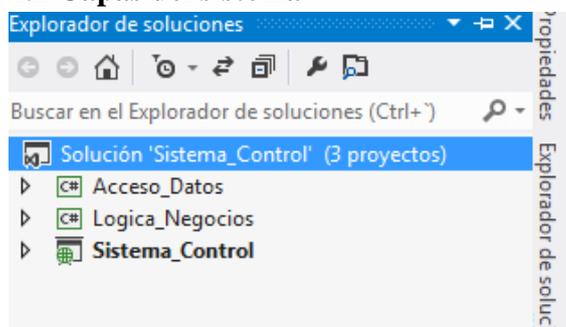


Figura 2: capas del sistema

### 1.3 Lógica de negocios, Clase usuario

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocios
{
    public class Logica_Usuario
    {
        public static DB_SISTEMADataContext dc = new DB_SISTEMADataContext();

        public static List<TBL_USUARIO> Lista_usuario()
        {
            var lista = dc.TBL_USUARIO.Where(usu =>
usu.USU_ESTADO.Equals("A"));
            return lista.ToList();
        }

        public static bool ingreso(string nombre, string clave)
        {
            var existe = dc.TBL_USUARIO.Any(exi =>
exi.USU_NOMBRE.Equals(nombre) & exi.USU_CLAVE.Equals(clave));
            return existe;
        }

        public static List<TBL_USUARIO> ObtenerUsuario()
        {
            var lista = dc.TBL_USUARIO.Where(usu =>
usu.USU_ESTADO.Equals("A"));
            return lista.ToList();
        }

        public static List<TBL_USUARIO> ObtenerUsuarioXIdList(int id)
        {
            var lista = dc.TBL_USUARIO.Where(usu => usu.USU_ESTADO.Equals("A")
&&
                usu.ID_USUARIO.Equals(id));
            return lista.ToList();
        }

        public static TBL_USUARIO ObtenerUsuarioXIdInfo(int id)
        {
            var usuario = dc.TBL_USUARIO.FirstOrDefault(usu =>
usu.USU_ESTADO.Equals("A") &&
                usu.ID_USUARIO.Equals(id));
            return usuario;
        }

        public static bool existe(string nom, string pass)
        {
            var existe = dc.TBL_USUARIO.Any(usu => usu.USU_ESTADO.Equals("A")
& usu.USU_NOMBRE.Equals(nom) & usu.USU_CLAVE.Equals(pass));
            return existe;
        }
    }
}
```

```
public static TBL_USUARIO ObtenerUsuarioXIdInfo(string nom, string
pass)
{
    var usuario = dc.TBL_USUARIO.FirstOrDefault(usu =>
usu.USU_ESTADO.Equals("A") &&
usu.USU_NOMBRE.Equals(nom) &
usu.USU_CLAVE.Equals(pass));
    return usuario;
}

//retorno de codigo
private static int IdCodigo()
{
    try
    {
        return dc.TBL_USUARIO.Max(usu => usu.ID_USUARIO + 1);
    }
    catch (Exception)
    {
        return 1;
    }
}

// save
public static void save(TBL_USUARIO usuarioInfo)
{
    try
    {
        usuarioInfo.ID_USUARIO = IdCodigo();
        usuarioInfo.USU_ESTADO = "A";
        //validacion(usuarioInfo);
        dc.TBL_USUARIO.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO GUARDADOS </br>" +
ex.Message);
    }
}

// modificar
public static void modify(TBL_USUARIO usuarioInfo)
{
    try
    {
        usuarioInfo.USU_ESTADO.Equals("A");
        //usuarioInfo.Id_Usuario = IdCodigo();
        //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS </br>" +
ex.Message);
    }
}

// eliminar
public static void delete(TBL_USUARIO usuarioInfo)
```

```
{
    try
    {
        usuarioInfo.USU_ESTADO = "E";
        //usuarioInfo.Id_Usuario = IdCodigo();
        //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO ELIMINADOS </br>" +
ex.Message);
    }
}

private static void validacion(Tbl_Usuario usuarioInfo)
{
    string mensaje = "";
    if (usuarioInfo.Id_Usuario == 0)
    {
        mensaje += "Ingrese tipo de usuario </br>";
    }
    else if (usuarioInfo.Usu_Apellido == "")
    {
        mensaje += "Ingrese el apellido";
    }
    else if (usuarioInfo.Usu_Nombre == "")
    {
        mensaje += "Ingrese el nombre";
    }
    else if (usuarioInfo.Usu_Cedula == "")
    {
        mensaje += "Ingrese la cedula";
    }
    else if
(!LogicaNegocios.utilidades.VerificacionCedula.CedulaCorrecta(usuarioInfo.Usu_
Cedula))
    {
        mensaje += "Cedula Incorrecta";
    }
    else if (usuarioInfo.Usu_Direccion == "")
    {
        mensaje += "Ingrese direccion";
    }
    else if (usuarioInfo.Usu_Telefono == "")
    {
        mensaje += "Ingrese telefono";
    }
    if (mensaje != "")
    {
        throw new ArgumentException(mensaje);
    }
}

public static TBL_USUARIO Autenticar(string nombre, string pass)
{
    var usuario = dc.TBL_USUARIO.FirstOrDefault(usu =>
usu.USU_ESTADO.Equals("A") & usu.USU_NOMBRE.Equals(nombre) &
usu.USU_CLAVE.Equals(pass));
}
```

```

        return usuario;
    }
}

```

## 1.4 Pagina Login

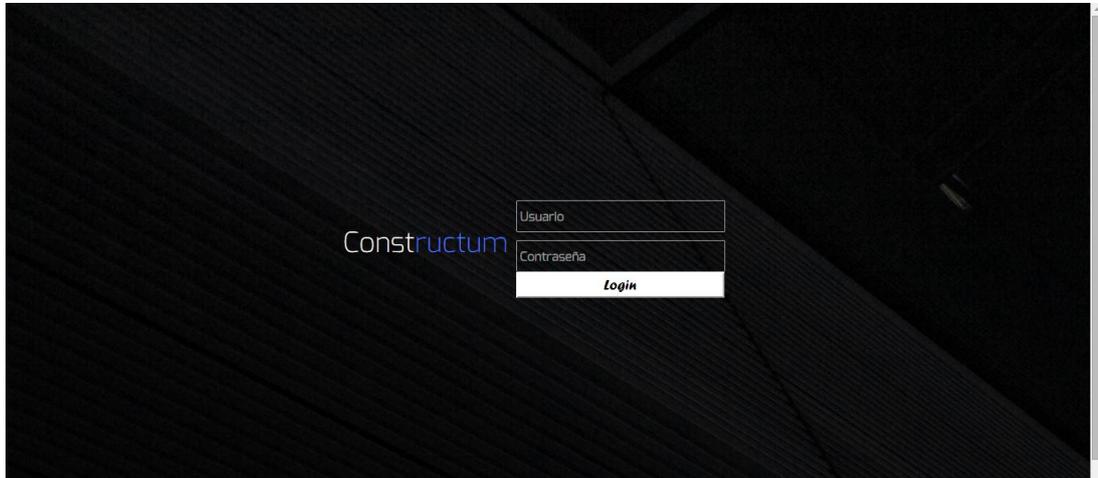


Figura 3: Modelo de la base de datos

### Código de la Página.

```

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs"
Inherits="Sistema_Control.Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

    <title>Random Login Form - CodePen</title>

    <style>
@import url(http://fonts.googleapis.com/css?family=Exo:100,200,400);
@import
url(http://fonts.googleapis.com/css?family=Source+Sans+Pro:700,400,300);
body{
    margin: 0;
    padding: 0;

    color: #fff;
    font-family: Arial;
    font-size: 12px;
}

.body{
    position: absolute;
    top: -20px;
    left: -20px;
    right: -40px;
    bottom: -40px;
    width: auto;

```

```
        height: auto;
        background-size: cover;
        -webkit-filter: blur(5px);
        z-index: 0;
    }

    .grad{
        position: absolute;
        top: -20px;
        left: -20px;
        right: -40px;
        bottom: -40px;
        width: auto;
        height: auto;
        background: -webkit-gradient(linear, left top, left bottom, color-
stop(0%,rgba(0,0,0,0)), color-stop(100%,rgba(0,0,0,0.65))); /* Chrome,Safari4+
*/
        z-index: 1;
        opacity: 0.7;
    }

    .header{
        position: absolute;
        top: calc(50% - 35px);
        left: calc(50% - 255px);
        z-index: 2;
    }

    .header div{
        float: left;
        color: #fff;
        font-family: 'Exo', sans-serif;
        font-size: 35px;
        font-weight: 200;
    }

    .header div span{
        color: #5379fa !important;
    }

    .login{
        position: absolute;
        top: calc(50% - 75px);
        left: calc(50% - 50px);
        height: 150px;
        width: 350px;
        padding: 10px;
        z-index: 2;
    }

    .login input[type=text]{
        width: 250px;
        height: 30px;
        background: transparent;
        border: 1px solid rgba(255,255,255,0.6);
        border-radius: 2px;
        color: #fff;
        font-family: 'Exo', sans-serif;
        font-size: 16px;
        font-weight: 400;
        padding: 4px;
    }
}
```

```
.login input[type=password]{
    width: 250px;
    height: 30px;
    background: transparent;
    border: 1px solid rgba(255,255,255,0.6);
    border-radius: 2px;
    color: #fff;
    font-family: 'Exo', sans-serif;
    font-size: 16px;
    font-weight: 400;
    padding: 4px;
    margin-top: 10px;
}

.login asp[type=button]{
    width: 260px;
    height: 35px;
    background: #fff;
    border: 1px solid #fff;
    cursor: pointer;
    border-radius: 2px;
    color: #a18d6c;
    font-family: 'Exo', sans-serif;
    font-size: 16px;
    font-weight: 400;
    padding: 6px;
    margin-top: 10px;
}

.login asp[type=button]:hover{
    opacity: 0.8;
}

.login asp[type=button]:active{
    opacity: 0.6;
}

.login input[type=text]:focus{
    outline: none;
    border: 1px solid rgba(255,255,255,0.9);
}

.login input[type=password]:focus{
    outline: none;
    border: 1px solid rgba(255,255,255,0.9);
}

.login asp[type=button]:focus{
    outline: none;
}

::-webkit-input-placeholder{
    color: rgba(255,255,255,0.6);
}

::-moz-input-placeholder{
    color: rgba(255,255,255,0.6);
}
}
</style>

<script src="js/prefixfree.min.js"></script>
```

```
</head>

<body background="images/fondo.jpg">

  <div class="body"></div>
  <div class="grad"></div>
  <div class="header">
    <div>Const<span>ructum</span></div>
  </div>
  <br>
  <div class="login">
    <form id="form1" runat="server">
      <label>
        <asp:TextBox ID="txt_log" runat="server"
placeholder="Usuario"></asp:TextBox>
      </label>
      <label>
        <asp:TextBox ID="txt_pass" runat="server"
placeholder="Contraseña" TextMode="Password"></asp:TextBox>
      </label>
      <br>
      <asp:Button ID="Button1" runat="server"
OnClick="Button1_Click" Text="Login" BackColor="White" BorderColor="White"
Font-Names="Forte" Font-Size="Large" Height="32px" Width="259px" />
      <br />
      <asp:Label ID="lblMensaje" runat="server"
Text="Label"></asp:Label>
    </form>
  </div>

</body>

</html>
```

### Código del Sistema

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Acceso_Datos;
using Logica_Negocios;
using System.Data.Linq;

namespace Sistema_Control
{
  public partial class Login : System.Web.UI.Page
  {
    TBL_USUARIO UserInfo = new TBL_USUARIO();
    protected void Page_Load(object sender, EventArgs e)
    {
      if (!IsPostBack)
      {
        lblMensaje.Visible = false;
      }
    }
  }
}
```

```
private void IngresoPage()
{
    string sms = "";
    if (txt_log.Text == "")
    {
        sms += "INGRESE NOMBRE </br>";
    }
    if (txt_pass.Text == "")
    {
        sms += "INGRESE PASSWORD </br>";
    }
    if (sms != "")
    {
        lblMensaje.Text = sms;
        lblMensaje.Visible = true;
        return;
    }

    string nombre = txt_log.Text;
    string clave = txt_pass.Text;

    bool existe = Logica_Usuario.existe(nombre, clave);
    if (existe)
    {
        UserInfo = new TBL_USUARIO();
        UserInfo = Logica_Usuario.Autenticar(nombre, clave);
        if (UserInfo != null)
        {
            Response.Redirect("~/Home.aspx", true);
        }
    }
    else
    {
        lblMensaje.Text = "USUARIO NO EXISTE";
        lblMensaje.Visible = true;
        return;
    }
}

protected void Button1_Click(object sender, EventArgs e)
{
    IngresoPage();
}
}
```

## 1.6 Lógica de negocios, Clase recurso

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Acceso_Datos;
using System.Data.Linq;

namespace Logica_Negocios
{
    public class Logica_Rekursos
    {
        public static DB_SISTEMADataContext dc = new DB_SISTEMADataContext();

        public static List<TBL_RECURSOS> Lista_rekursos_mat()
        {
            var lista = dc.TBL_RECURSOS.Where(rec =>
rec.REC_ESTADO.Equals("A") && rec.ID_TIPORECURSOS.Equals(2));
            return lista.ToList();
        }

        public static List<TBL_RECURSOS> Lista_rekursos_herr()
        {
            var lista = dc.TBL_RECURSOS.Where(rec =>
rec.REC_ESTADO.Equals("A") && rec.ID_TIPORECURSOS.Equals(1));
            return lista.ToList();
        }

        public static List<TBL_RECURSOS> Lista_rekursos_matXBodega(int bodega)
        {
            var lista = dc.TBL_RECURSOS.Where(rec =>
rec.REC_ESTADO.Equals("A") && rec.ID_TIPORECURSOS.Equals(2) &&
rec.ID_BODEGA.Equals(bodega));
            return lista.ToList();
        }

        public static List<TBL_RECURSOS> Lista_rekursos_herrXBodega(int
bodega)
        {
            var lista = dc.TBL_RECURSOS.Where(rec =>
rec.REC_ESTADO.Equals("A") && rec.ID_TIPORECURSOS.Equals(1) &&
rec.ID_BODEGA.Equals(bodega));
            return lista.ToList();
        }

        public static TBL_RECURSOS ObtenerRecursoXIdInfo(int id)
        {
            var usuario = dc.TBL_RECURSOS.FirstOrDefault(rec =>
rec.REC_ESTADO.Equals("A") &&
rec.ID_RECURSOS.Equals(id));
            return usuario;
        }

        //retorno de codigo
        private static int IdCodigo()
        {
            try
            {
                return dc.TBL_RECURSOS.Max(rec => rec.ID_RECURSOS + 1);
            }
        }
    }
}
```

```
        catch (Exception)
        {
            return 1;
        }
    }

    // save
    public static void save(TBL_RECURSOS usuarioInfo)
    {
        try
        {
            usuarioInfo.REC_ESTADO = "A";
            usuarioInfo.ID_RECURSOS = IdCodigo();
            //validacion(usuarioInfo);
            dc.TBL_RECURSOS.InsertOnSubmit(usuarioInfo);
            dc.SubmitChanges();
        }
        catch (Exception ex)
        {

            throw new ArgumentException("DATOS NO GUARDADOS </br>" +
ex.Message);
        }
    }

    // modificar
    public static void modify(TBL_RECURSOS usuarioInfo)
    {
        try
        {
            usuarioInfo.REC_ESTADO="A";
            //usuarioInfo.Id_Usuario = IdCodigo();
            //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
            dc.SubmitChanges();
        }
        catch (Exception ex)
        {

            throw new ArgumentException("DATOS NO MODIFICADOS </br>" +
ex.Message);
        }
    }

    // eliminar
    public static void delete(TBL_RECURSOS usuarioInfo)
    {
        try
        {
            usuarioInfo.REC_ESTADO = "E";
            //usuarioInfo.Id_Usuario = IdCodigo();
            //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
            dc.SubmitChanges();
        }
        catch (Exception ex)
        {

            throw new ArgumentException("DATOS NO ELIMINADOS </br>" +
ex.Message);
        }
    }
}
```

```
}  
}
```

## 1.5 Pagina para recursos

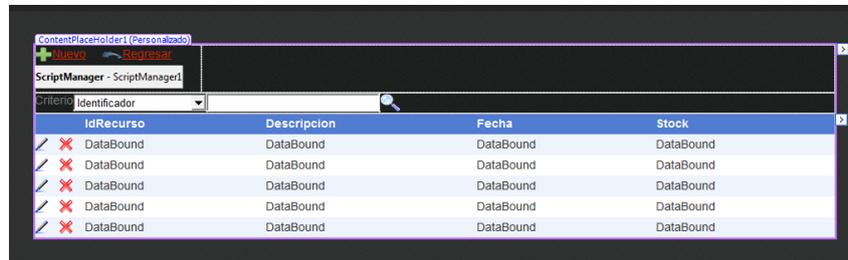


Figura 4: diseño de la página recursos

### Código de la página

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin.Master"  
AutoEventWireup="true" CodeBehind="Herramientas.aspx.cs"  
Inherits="Sistema_Control.Sistema.Bodega.Herramientas.Herramientas" %>  
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">  
</asp:Content>  
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"  
runat="server">  
  <table width="100%" border="0" cellpadding="0" cellspacing="0">  
    <tr>  
      <td>  
        <table width="200px">  
          <tr>  
            <td colspan="2">  
              <div style="margin-bottom: 5px;">  
                <asp:ImageButton ID="imgNuevo" runat="server"  
ImageUrl="~/images/iconos/plus.png" Height="18px" Width="21px"  
OnClick="imgNuevo_Click"/>  
                <asp:LinkButton ID="lnkNuevo" runat="server"  
OnClick="lnkNuevo_Click"> Nuevo</asp:LinkButton>  
                <asp:ImageButton ID="imgAtras" runat="server"  
ImageUrl="~/images/iconos/Atras.png"  
Height="19px" Width="25px"  
style="margin-left: 19px"  
OnClick="imgAtras_Click" />  
                <asp:LinkButton ID="lnkAtras" runat="server"  
OnClick="lnkAtras_Click" >Regresar</asp:LinkButton>  
              <asp:ScriptManager ID="ScriptManager1"  
runat="server">  
                </asp:ScriptManager>  
            </div>  
          </td>  
        </tr>  
      </table>  
    </td>  
  </tr>  
</table>  
</td>  
</tr>  
<tr>  
  <td>
```

```

        <table width="100%">
            <tr>
                <td>
                    <asp:UpdatePanel ID="ip_update2" runat="server">
                        <ContentTemplate>
                            <asp:Label ID="label1" runat="server"
Text="Criterio"></asp:Label>
                            <asp:DropDownList ID="ddl_Criterio" runat="server"
Width="160px" AutoPostBack="true"
OnSelectedIndexChanged="ddl_Criterio_SelectedIndexChanged">
                                <asp:ListItem Value="id"
Text="Identificador"></asp:ListItem>
                                <asp:ListItem Value="nom"
Text="Nombre"></asp:ListItem>
                            </asp:DropDownList>
                            <asp:TextBox ID="txt_Busqueda" runat="server"
Width="200px"></asp:TextBox>
                            <asp:ImageButton ID="imgBuscar" runat="server"
ImageUrl="~/images/iconos/Search 2.png" Height="23px" Width="26px"
OnClick="imgBuscar_Click" />
                        </ContentTemplate>
                    </asp:UpdatePanel>
                </td>
            </tr>
        </table>
    </td>
</tr>
<tr>
    <td>
        <asp:UpdatePanel ID="updatePnl" runat="server">
            <ContentTemplate>
                <asp:DataGrid ID="Dgr_HerramientasLista"
runat="server" CellPadding="4" ForeColor="#333333" Width="100%"
GridLines="None" AutoGenerateColumns="False"
OnItemCommand="Dgr_HerramientasLista_ItemCommand">
                    <AlternatingItemStyle BackColor="White" />
                    <EditItemStyle BackColor="#2461BF" />
                    <FooterStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                    <HeaderStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                    <ItemStyle BackColor="#EFF3FB" />
                    <PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
                    <SelectedItemStyle BackColor="#D1DDF1" Font-
Bold="True" ForeColor="#333333" />
                    <Columns>
                        <asp:TemplateColumn ItemStyle-Width="30px">
                            <ItemTemplate>
                                <asp:ImageButton ID="imgEditar"
runat="server" CausesValidation="false" CommandName="Editar"
ImageUrl
="~/images/iconos/Highlight.png" ToolTip="Editar" Width="16" />
                            </ItemTemplate>
                            <EditItemTemplate>
                                <asp:ImageButton ID="imgActualizar"
runat="server" CausesValidation="false" CommandName="Grabar"
ImageUrl="~/images/iconos/Save.png"
ToolTip="Actualizar" Width="16" />
                            </EditItemTemplate>
                        </asp:TemplateColumn>
                    </Columns>
                </asp:DataGrid>
            </ContentTemplate>
        </asp:UpdatePanel>
    </td>
</tr>

```

```

        </asp:TemplateColumn>
        <asp:TemplateColumn ItemStyle-Width="30px">
            <ItemTemplate>
                <asp:ImageButton ID="imgEliminar"
runat="server" CausesValidation="false" CommandName='Eliminar'
                ImageUrl="~/images/iconos/Error.png"
runat="server" CausesValidation="false" CommandName="Cancelar"
                ImageUrl="~/images/iconos/Alvertencia.png" Tooltip="Cancelar" Width="16" />
            </ItemTemplate>
            <EditItemTemplate>
                <asp:ImageButton ID="imgCancelar"
runat="server" CausesValidation="false" CommandName="Cancelar"
                ImageUrl="~/images/iconos/Alvertencia.png" Tooltip="Cancelar" Width="16" />
            </EditItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn
HeaderText="IdRecurso">
            <ItemTemplate>
                <span style="float:left;">
                    <asp:Label ID="lbl_idHerramienta"
runat="server" Text='<%#Eval("ID_RECURSOS") %>'></asp:Label>
                </span>
            </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="Descripcion">
            <ItemTemplate>
                <span style="float:left;">
                    <asp:Label ID="lbl_Descripcion"
runat="server" Text='<%#Eval("REC_DESCRIPCION") %>'></asp:Label>
                </span>
            </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="Fecha">
            <ItemTemplate>
                <span style="float:left;">
                    <asp:Label ID="lbl_Fecha"
runat="server" Text='<%#Eval("REC_FECHACREACION") %>'></asp:Label>
                </span>
            </ItemTemplate>
        </asp:TemplateColumn>
        <asp:TemplateColumn HeaderText="Stock">
            <ItemTemplate>
                <span style="float:left;">
                    <asp:Label ID="lbl_stock"
runat="server" Text='<%#Eval("REC_STOCK") %>'></asp:Label>
                </span>
            </ItemTemplate>
        </asp:TemplateColumn>
    </Columns>
</asp:DataGrid>
</ContentTemplate>
</asp:UpdatePanel>
</td>
</tr>
</table>
</asp:Content>

using System;
using System.Collections.Generic;

```

```

using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Acceso_Datos;
using Logica_Negocios;
using System.Data.Linq;

namespace Sistema_Control.Sistema.Bodega.Herramientas
{
    public partial class Herramientas : System.Web.UI.Page
    {
        TBL_RECURSOS userinfo = new TBL_RECURSOS();
        List<TBL_RECURSOS> userList = new List<TBL_RECURSOS>();
        int codigo;
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                if (Request["IdBodega"] != null)
                {
                    codigo = Convert.ToInt32(Request["IdBodega"]);
                }
                CargarUsuario();
            }
            else
            {
                codigo = Convert.ToInt32(Request["IdBodega"]);
            }
        }
        private void CargarUsuario()
        {
            userList = Logica_Recursos.Lista_recursos_herrXBodega(codigo);
            if (userList != null)
            {
                Dgr_HerramientasLista.DataSource = userList;
                Dgr_HerramientasLista.DataBind();
            }
        }

        protected void Dgr_HerramientasLista_ItemCommand(object source,
        DataGridCommandEventArgs e)
        {
            try
            {
                int idcodigo =
                Int32.Parse(((Label)e.Item.Cells[2].FindControl("lbl_idHerramienta")).Text);

                if (e.CommandName == "Editar")
                {
                    Response.Redirect("~/Sistema/Bodega/Herramientas/DetalleHerramienta.aspx?IdRec
                    urso=" + idcodigo.ToString(), true);
                }

                if (e.CommandName == "Eliminar")
                {
                    userinfo = new TBL_RECURSOS();
                    userinfo =
                    Logica_Recursos.ObtenerRecursoXIdInfo(idcodigo);
                }
            }
        }
    }
}

```

```
        if (userinfo != null)
        {
            Logica_Recursos.delete(userinfo);
            CargarUsuario();
        }
    }
}
catch
{
}
}

private void nuevo()
{
    int idcodigo = codigo;

Response.Redirect("~/Sistema/Bodega/Herramientas/DetalleHerramienta.aspx?IdBodega=" + idcodigo.ToString(), true);
}

protected void imgNuevo_Click(object sender, ImageClickEventArgs e)
{
    nuevo();
}

protected void lnkNuevo_Click(object sender, EventArgs e)
{
    nuevo();
}
private void regreso()
{
    int idcodigo = codigo;
    Response.Redirect("~/Sistema/Bodega/BodegaNuevo.aspx?IdBodega=" +
idcodigo.ToString(), true);
}
protected void lnkAtras_Click(object sender, EventArgs e)
{
    regreso();
}

protected void imgAtras_Click(object sender, ImageClickEventArgs e)
{
    regreso();
}

protected void ddl_Criterio_SelectedIndexChanged(object sender,
EventArgs e)
{
    string op = ddl_Criterio.SelectedValue;
    switch (op)
    {
        case "id":
            txt_Busqueda.Visible = false;
            imgBuscar.Visible = true;
            break;
        case "nom":
            txt_Busqueda.Visible = true;
            imgBuscar.Visible = false;
            break;
        default:
            break;
    }
}
```

```
    }  
  }  
}
```

## 1.6 Lógica de negocios, Clase Ingreso\_Egreso

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using Acceso_Datos;  
using System.Data.Linq;  
  
namespace Logica_Negocios  
{  
    public class Logica_Ingreso_Egreso  
    {  
        public static DB_SISTEMADataContext dc = new DB_SISTEMADataContext();  
  
        public static List<TBL_INGRESO_EGRESO> Lista_ingreso_egreso()  
        {  
            var lista = dc.TBL_INGRESO_EGRESO.Where(ine =>  
ine.INE_ESTADO.Equals("A"));  
            return lista.ToList();  
        }  
        public static List<TBL_INGRESO_EGRESO> Lista_ingreso_egresoXid(int  
codigo)  
        {  
            var lista = dc.TBL_INGRESO_EGRESO.Where(ine =>  
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(codigo));  
            return lista.ToList();  
        }  
        public static TBL_INGRESO_EGRESO ObtenerUsuarioXIdInfo(int id)  
        {  
            var usuario = dc.TBL_INGRESO_EGRESO.FirstOrDefault(ine =>  
ine.INE_ESTADO.Equals("A") &&  
ine.ID_INGRESOEGRESO.Equals(id));  
            return usuario;  
        }  
  
        public static List<TBL_INGRESO_EGRESO> Lista_ingresoXid(int id)  
        {  
            var lista = dc.TBL_INGRESO_EGRESO.Where(ine =>  
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(id) &&  
ine.ID_TIPOIE.Equals(1));  
            return lista.ToList();  
        }  
  
        public static List<TBL_INGRESO_EGRESO> Lista_salidasXid(int id)  
        {  
            var lista = dc.TBL_INGRESO_EGRESO.Where(ine =>  
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(id) &&  
ine.ID_TIPOIE.Equals(2));  
            return lista.ToList();  
        }  
  
        public static int ObtenerCantidadBodega(int idrecurso)
```

```
{
    int a = IdCodigo();
    a = a - 1;
    int entradas=0;
    int salidas = 0;
    for (int i = 0; i <= a; i++)
    {
        var lista = dc.TBL_INGRESO_EGRESO.FirstOrDefault(ine =>
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(idrecurso) &&
ine.ID_TIPOIE.Equals(1) && ine.ID_INGRESOEGRESO.Equals(i));
        if (lista != null)
        {
            entradas = entradas + Convert.ToInt32(lista.INE_CANTIDAD);
        }
    }

    for (int i = 0; i <= a; i++)
    {
        var lista = dc.TBL_INGRESO_EGRESO.FirstOrDefault(ine =>
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(idrecurso) &&
ine.ID_TIPOIE.Equals(2) && ine.ID_INGRESOEGRESO.Equals(i));
        if (lista != null)
        {
            salidas = salidas + Convert.ToInt32(lista.INE_CANTIDAD);
        }
    }
    int total = entradas - salidas ;
    return total;
}

public static int ObtenerCantidadTotal(int idrecurso)
{
    int a = IdCodigo();
    a = a - 1;
    int entradas = 0;
    int salidas = 0;
    for (int i = 0; i <= a; i++)
    {
        var lista = dc.TBL_INGRESO_EGRESO.FirstOrDefault(ine =>
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(idrecurso) &&
ine.ID_TIPOIE.Equals(1) && ine.ID_INGRESOEGRESO.Equals(i));
        if (lista != null)
        {
            entradas = entradas + Convert.ToInt32(lista.INE_CANTIDAD);
        }
    }

    for (int i = 0; i <= a; i++)
    {
        var lista = dc.TBL_INGRESO_EGRESO.FirstOrDefault(ine =>
ine.INE_ESTADO.Equals("A") && ine.ID_RECURSOS.Equals(idrecurso) &&
ine.ID_TIPOIE.Equals(2) && ine.ID_INGRESOEGRESO.Equals(i));
        if (lista != null)
        {
            salidas = salidas + Convert.ToInt32(lista.INE_CANTIDAD);
        }
    }
    int total = entradas + salidas;
    return total;
}
```

```
//retorno de codigo
private static int IdCodigo()
{
    try
    {
        return dc.TBL_INGRESO_EGRESO.Max(ine => ine.ID_INGRESOEGRESO +
1);
    }
    catch (Exception)
    {
        return 1;
    }
}

// save
public static void save(TBL_INGRESO_EGRESO usuarioInfo)
{
    try
    {
        usuarioInfo.INE_ESTADO = "A";
        usuarioInfo.ID_INGRESOEGRESO = IdCodigo();
        //validacion(usuarioInfo);
        dc.TBL_INGRESO_EGRESO.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO GUARDADOS </br>" +
ex.Message);
    }
}

// modificar
public static void modify(TBL_INGRESO_EGRESO usuarioInfo)
{
    try
    {
        usuarioInfo.INE_ESTADO.Equals('A');
        //usuarioInfo.Id_Usuario = IdCodigo();
        //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS </br>" +
ex.Message);
    }
}

// eliminar
public static void delete(TBL_INGRESO_EGRESO usuarioInfo)
{
    try
    {
        usuarioInfo.INE_ESTADO = "E";
        //usuarioInfo.Id_Usuario = IdCodigo();
        //dc.Tbl_Usuario.InsertOnSubmit(usuarioInfo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
```

```

        {
            throw new ArgumentException("DATOS NO ELIMINADOS </br>" +
ex.Message);
        }
    }
}
}

```

## 1.7 Página historial



IdBodega	Descripción	Fecha	Cantidad
✘ DataBound	DataBound	DataBound	DataBound
✘ DataBound	DataBound	DataBound	DataBound
✘ DataBound	DataBound	DataBound	DataBound
✘ DataBound	DataBound	DataBound	DataBound
✘ DataBound	DataBound	DataBound	DataBound

Figura 5: diseño de la página historial

Código de la página

```

<%@ Page Title="" Language="C#" MasterPageFile="~/Admin.Master"
AutoEventWireup="true" CodeBehind="Historial.aspx.cs"
Inherits="Sistema_Control.Sistema.Bodega.Ingreso_Egreso.Historial" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
    <strong class="title-1">Menu Bodega</strong>
    <br />
    <table width="100%" border="0" cellpadding="0" cellspacing="0">
        <tr>
            <td>
                <table width="200px">
                    <tr>
                        <td colspan="2">
                            <div style="margin-bottom: 5px;">
                                <asp:ImageButton ID="imgNuevo" runat="server"
ImageUrl="~/images/iconos/plus.png" Height="18px" Width="21px" />
                                <asp:LinkButton ID="lnkNuevo" runat="server"
> Nuevo</asp:LinkButton>
                                <asp:ScriptManager ID="ScriptManager1"
runat="server">
                                    </asp:ScriptManager>
                            </div>
                        </td>
                    </tr>
                </table>
            </td>
            <td>
                <table width="100%">
                    <tr>
                        <td>
                            <asp:Label ID="label1" runat="server"
Text="Criterio"></asp:Label>

```

```

        <asp:DropDownList ID="ddl_Criterio" runat="server"
Width="160px">
            <asp:ListItem Value="id"
Text="Codigo"></asp:ListItem>
            <asp:ListItem Value="Nom"
Text="Nombre"></asp:ListItem>
        </asp:DropDownList>
        <asp:TextBox ID="txt_Busqueda" runat="server"
Width="200px"></asp:TextBox>
        <asp:ImageButton ID="imgBuscar" runat="server"
ImageUrl="~/images/iconos/Search 2.png" Height="23px" Width="26px" />
    </td>
</tr>
</table>
</td>
</tr>
<tr>
<td>
        <asp:UpdatePanel ID="updatePnl" runat="server">
            <ContentTemplate>
                <asp:DataGrid ID="Dgr_IE" runat="server"
CellPadding="4" ForeColor="#333333" Width="100%" GridLines="None"
AutoGenerateColumns="False" OnItemCommand="Dgr_IE_ItemCommand">
                    <AlternatingItemStyle BackColor="White" />
                    <EditItemStyle BackColor="#2461BF" />
                    <FooterStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                    <HeaderStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                    <ItemStyle BackColor="#EFF3FB" />
                    <PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
                    <SelectedItemStyle BackColor="#D1DDF1" Font-
Bold="True" ForeColor="#333333" />
                    <Columns>
                        <asp:TemplateColumn ItemStyle-Width="30px">
                            <ItemTemplate>
                                <asp:ImageButton ID="imgEditar"
runat="server" CausesValidation="false" CommandName="Editar"
                                ImageUrl
                                =~/images/iconos/Highlight.png" Tooltip="Editar" Width="16" />
                            </ItemTemplate>
                            <EditItemTemplate>
                                <asp:ImageButton ID="imgActualizar"
runat="server" CausesValidation="false" CommandName="Grabar"
                                ImageUrl="~/images/iconos/Save.png"
                                Tooltip="Actualizar" Width="16" />
                            </EditItemTemplate>
                        </asp:TemplateColumn>
                        <asp:TemplateColumn ItemStyle-Width="30px">
                            <ItemTemplate>
                                <asp:ImageButton ID="imgEliminar"
runat="server" CausesValidation="false" CommandName='Eliminar'
                                ImageUrl="~/images/iconos/Error.png"
                                Tooltip='' Width="16" OnClientClick="return confirm('Seguro que desea eliminar
                                el registro')" />
                            </ItemTemplate>
                            <EditItemTemplate>
                                <asp:ImageButton ID="imgCancelar"
runat="server" CausesValidation="false" CommandName="Cancelar"
                                ImageUrl="~/images/iconos/Alvertencia.png" Tooltip="Cancelar" Width="16" />

```

```
        </EditItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn
HeaderText="IdBodega">
        <ItemTemplate>
            <span style="float:left;">
                <asp:Label ID="lbl_idIE"
runat="server" Text='<%#Eval("ID_INGRESOEGRESO") %>'></asp:Label>
            </span>
        </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="Descripcion">
        <ItemTemplate>
            <span style="float:left;">
                <asp:Label ID="lbl_Descripcion"
runat="server" Text='<%#Eval("INE_DESCRIPCION") %>'></asp:Label>
            </span>
        </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="Fecha">
        <ItemTemplate>
            <span style="float:left;">
                <asp:Label ID="lbl_Fecha"
runat="server" Text='<%#Eval("INE_FECHA") %>'></asp:Label>
            </span>
        </ItemTemplate>
    </asp:TemplateColumn>
    <asp:TemplateColumn HeaderText="Cantidad">
        <ItemTemplate>
            <span style="float:left;">
                <asp:Label ID="lbl_Cantidad"
runat="server" Text='<%#Eval("INE_CANTIDAD") %>'></asp:Label>
            </span>
        </ItemTemplate>
    </asp:TemplateColumn>
</Columns>
</asp:DataGrid>
</ContentTemplate>
</asp:UpdatePanel>
</td>
</tr>
</table>
</asp:Content>
```

Código del sistema

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Acceso_Datos;
using Logica_Negocios;
using System.Data.Linq;

namespace Sistema_Control.Sistema.Bodega.Ingreso_Egreso
{
    public partial class Historial : System.Web.UI.Page
    {
        TBL_INGRESO_EGRESO userinfo = new TBL_INGRESO_EGRESO();
    }
}
```

```
List<TBL_INGRESO_EGRESO> userList = new List<TBL_INGRESO_EGRESO>();

TBL_DETALLE DetalleInfo = new TBL_DETALLE();
TBL_RECURSOS RecursoInfo = new TBL_RECURSOS();
int codigo;
protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        if (Request["IdRecurso"] != null)
        {
            codigo = Convert.ToInt32(Request["IdRecurso"]);
        }
        CargarUsuario();
    }
    else
    {
        codigo = Convert.ToInt32(Request["IdRecurso"]);
    }
}
private void CargarUsuario()
{
    userList = Logica_Ingreso_Egreso.Lista_ingreso_egresoXid(codigo);
    if (userList != null)
    {
        Dgr_IE.DataSource = userList;
        Dgr_IE.DataBind();
    }
}

protected void Dgr_IE_ItemCommand(object source,
DataGridCommandEventArgs e)
{
    try
    {
        int idcodigo =
Int32.Parse(((Label)e.Item.Cells[2].FindControl("lbl_idIE")).Text);
        if (e.CommandName == "Editar")
        {
            TBL_INGRESO_EGRESO IEInfo = new TBL_INGRESO_EGRESO();
            IEInfo =
Logica_Ingreso_Egreso.ObtenerUsuarioXidInfo(idcodigo);
            int val = Convert.ToInt32(IEInfo.ID_TIPOIE);

            if (val == 1)
            {
                Response.Redirect("~/Sistema/Bodega/Ingreso_Egreso/Entradas.aspx?IdRecurso=" +
codigo.ToString()+"&aux="+idcodigo.ToString(), true);
            }
            else
            {
                Response.Redirect("~/Sistema/Bodega/Ingreso_Egreso/Salidas.aspx?IdRecurso=" +
idcodigo.ToString(), true);
            }
        }
        if (e.CommandName == "Eliminar")
        {

```

```
        userinfo = new TBL_INGRESO_EGRESO();
        userinfo =
Logica_Ingreso_Egreso.ObtenerUsuarioXIdInfo(idcodigo);
        if (userinfo != null)
        {
            Logica_Ingreso_Egreso.delete(userinfo);
            Totales();
            CargarUsuario();
        }
    }
}
catch
{
}
}
int val;
private void Totales()
{
    int totalBodega =
Logica_Ingreso_Egreso.ObtenerCantidadBodega(codigo);
    int totalStock =
Logica_Ingreso_Egreso.ObtenerCantidadTotal(codigo);
    TBL_DETALLE RecursoId = new TBL_DETALLE();
    RecursoId = Logica_Detalle.ObtenerUsuarioXIdInfo(codigo);
    decimal costoUnit = Convert.ToDecimal(RecursoId.DET_COSTOUNIT);
    decimal costoTotal = totalBodega * costoUnit;

    RecursoInfo = Logica_Recursos.ObtenerRecursoXIdInfo(codigo);
    RecursoInfo.REC_STOCK = totalStock;
    Logica_Recursos.modify(RecursoInfo);
    val = Convert.ToInt32(RecursoInfo.ID_TIPORECURSOS);

    DetalleInfo = Logica_Detalle.ObtenerUsuarioXIdInfo(codigo);
    DetalleInfo.DET_COSTOTOTAL = costoTotal;
    DetalleInfo.DET_TOTAL = totalBodega;
    Logica_Detalle.modify(DetalleInfo);
}
}
}
```

## 1.8 Script QR

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="index.aspx.cs"
Inherits="pp.index" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/jquery.webcamqrcode.min.js"></script>
<title></title>
</head>
<body>
    <form id="form1" runat="server">
```

```
WebcamQRCode View:<br />
<script language="javascript" type="text/javascript">

    (function ($) {
        $('document').ready(function () {
            $('#qrcodebox').WebcamQRCode({
                onQRCodeDecode: function (p_data) {
                    //$('#qrcode_result').html(p_data);
                    $("#<%=qrcode_result.ClientID%>").val(p_data);
                }
            });

            $('#btn_start').click(function () {
                $('#qrcodebox').WebcamQRCode().start();
            });

            $('#btn_stop').click(function () {
                $('#qrcodebox').WebcamQRCode().stop();
            });
        });
    })(jQuery);
</script>
<div style="width: 350px; height: 350px;" id="qrcodebox">
</div>
<input type="button" value="Start" id="btn_start" />
<input type="button" value="Stop" id="btn_stop" />
<p>
    <asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>
    Last QRCode value:
    <!--<span id="qrcode_result">none</span>-->
    <asp:TextBox ID="qrcode_result" Visible="true"
    runat="server">none</asp:TextBox>

</p>
    <p>
        <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
    Text="Button" />
    </p>
</form>
</body>
```

## GLOSARIO DE TÉRMINOS

**Sistema.**-Es un conjunto de partes o elementos, organizadas y relacionadas que interactúan entre sí para lograr un objetivo.

**Interfaces.**-Es un elemento de conexión que facilita el intercambio de datos.

**Aspx.**- Es un lenguaje de código diseñado para el desarrollo web y que puede ser incrustado en HTML.

**HTML.**- Es un lenguaje de programación que se utiliza para el desarrollo de páginas de Internet.

**Clase de asociación.**- describe una relación entre dos clases o entre las instancias de dos clases.

**Línea de asociación.**- En el Diseñador de clases, es una línea que muestra que dos tipos están relacionados.

**Conjunto de asociaciones.**- En el marco de entidades de ADO.NET, un contenedor lógico para las instancias de asociaciones del mismo tipo.

**Clase.**- Tipo de referencia que encapsula datos (constantes y campos) y el comportamiento (métodos, propiedades, indizadores, eventos, operadores, constructores de instancia, constructores estáticos y destructores), y puede contener tipos anidados.

**Clase.**- Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos,<sup>3</sup> basado en prototipos, imperativo, débilmente tipado y dinámico.



INSTITUTO TECNOLÓGICO  
"CORDILLERA"

CARRERA ANÁLISIS DE SISTEMAS

CONTROL DE PROCESOS DE INVENTARIO Y NÓMINA MEDIANTE UN  
SISTEMA WEB PARA LA EMPRESA CONSTRUCTUM S. A UBICADA EN LA  
CIUDAD DE QUITO

Manual de instalación

Autor: Peñafiel Asimbaya Bryan Patricio

Tutor: Ing. Hugo Heredia

Quito, Octubre 2014



## Índice

Instalación Visual Studio Ultimate 2012.....	146
Figura 1 Instalador Visual Studio Ultimate 2012.....	146
Figura 2 Términos y condiciones de licencia.....	146
Figura 3 Complementos en la instalación de Visual Studio 2012.....	147
Figura 4 avance de instalación .....	147
Figura 5 Clave del producto .....	148
Figura 6 Instalación finalizada .....	148
Figura 7 Configuración de entorno predeterminada.....	149
Figura 8 Inicio Visual 2012.....	149

## Instalación Visual Studio Ultimate 2012

Ejecutamos el instalador dándole clic en el botón si

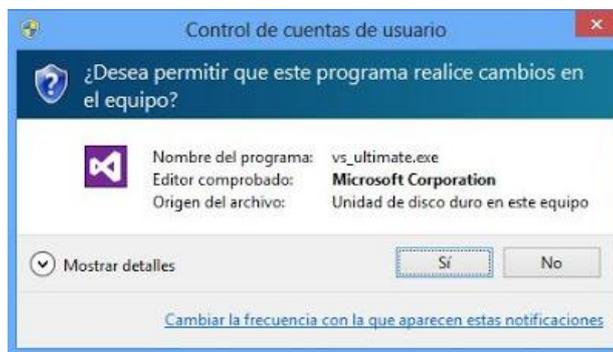


Figura 1 Instalador Visual Studio Ultimate 2012

Leemos los términos de licencia si estamos de acuerdo damos clic marcando Acepto los términos y condiciones de la licencia



Figura 2 Términos y condiciones de licencia

Marcaremos las siguientes características a instalar



Figura 3 Complementos en la instalación de Visual Studio 2012

El asistente de instalación indica que todo se encuentra correcto

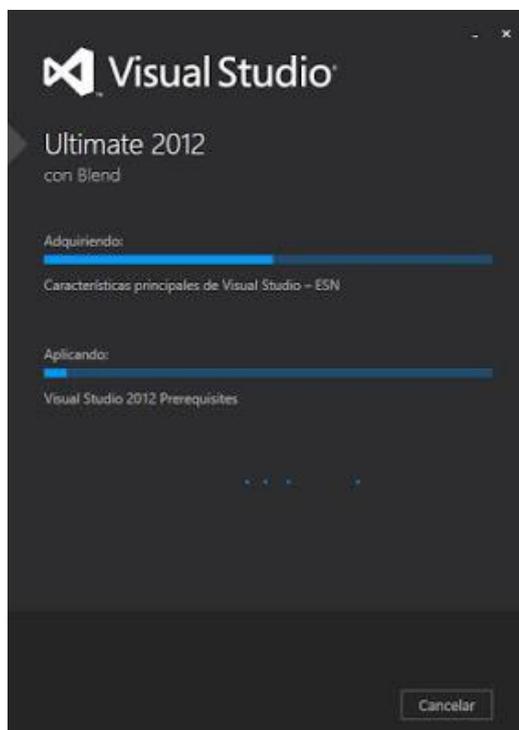


Figura 4 avance de instalación

Visual Studio Ultimate 2012 solicitará que introduzcamos la clave del producto de lo contrario se activará la prueba de 90 días y terminará la instalación



Figura 5 Clave del producto

Instalación exitosa y completa



Figura 6 Instalación finalizada

Primera entrada al sistema se inicia configuración de entorno predeterminado damos clic a iniciar Visual Studio

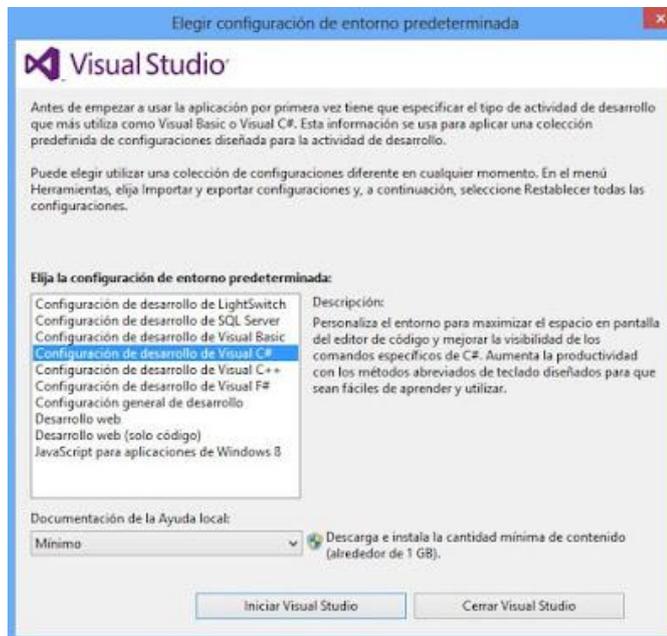


Figura 7 Configuración de entorno predeterminada

Interfaz de inicio de Visual Studio 2012

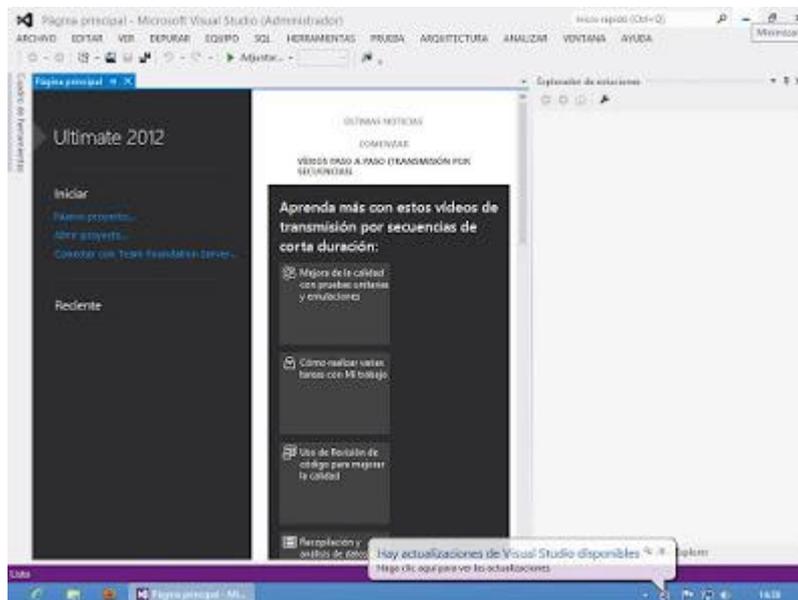


Figura 8 Inicio Visual 2012

## Bibliografía

<http://www.monografias.com/trabajos11/conin/conin.shtml>

<http://es.wikipedia.org/wiki/N%C3%B3mina>

<http://www.asp.net/>

[http://es.wikipedia.org/wiki/C%C3%B3digo\\_QR](http://es.wikipedia.org/wiki/C%C3%B3digo_QR)

Utilización de UML en Ingeniería del Software con Objetos y Componentes. Perditá Stevens, Rob Pooley. Addison Wesley. 2002.

UML 2 Perditá Stevens Pearson Education ISBN-10: 8478290869