



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 4

APLICACIÓN WEB PARA LA MEJORA EN LA GESTIÓN DE SERVICIOS DE TRANSCONSUL S.A.

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor: Téc. Carlos Brayan Rámila Chorens

Tutora: MSc. Yadira Ramírez Rodríguez

La Habana, 2024

La eficaz gestión de citas y servicios legales no solo es una necesidad operativa, sino un pilar fundamental para garantizar la confianza y satisfacción de nuestros clientes en cada interacción.

John Smith, Experto en Gestión Legal

A mi familia, amigos, compañeros y profesores, quienes con su amor, apoyo, sabiduría y aliento han sido mi mayor inspiración y motivación en cada paso del camino. Este trabajo está dedicado a ustedes, quienes han hecho posible este logro con su constante presencia y ayuda incondicional.

Agradecimientos

Quiero expresar mi más profundo agradecimiento a todas las personas que han sido parte de este viaje académico. A mi familia, por su amor incondicional y su constante apoyo en cada desafío que he enfrentado. A mis amigos, por estar siempre presentes, brindándome su amistad y aliento en los momentos difíciles. A mis compañeros, por compartir este camino conmigo, por los momentos de colaboración y aprendizaje mutuo. Y a mis profesores, por su orientación, sabiduría y dedicación, que han sido fundamentales para mi crecimiento académico y personal.

En especial, quiero agradecer a mi tutora y profesora, MSc. Yadira Ramírez Rodríguez, por su invaluable guía y apoyo durante todo el proceso de investigación y redacción de esta tesis. Su paciencia, experiencia y compromiso han sido fundamentales para alcanzar este logro.

Por último, pero no menos importante, quiero agradecerme a mí mismo por mi perseverancia, determinación y dedicación para alcanzar mis metas. Sin mi propia fuerza interior y convicción, este logro no habría sido posible.

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Técn. Carlos Brayan Rámila Chorens
Autor

MSc. Yadirá Ramírez Rodríguez
Tutora

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna.

Palabras clave: word1, word2, word3.

Introducción	1
1 Fundamentación Teórica	4
1.1 Introducción al capítulo	4
1.2 Sistema de gestión de citas y servicios	4
1.3 Sistemas Homólogos	4
1.4 Metodología de desarrollo de software	4
1.5 Herramientas y tecnología de desarrollo de software	5
1.6 Sección de prueba de la UCI	7
2 Propuesta de solución	9
2.1 Introducción al capítulo	9
2.2 Propuesta de solución	9
2.3 Fase I: Planificación	10
2.4 Sección de prueba	13
2.5 Ejemplos de código fuente	14
3 Ejemplo de tabla grande	20
3.1 Tabla aleatoria	20
3.2 Tablas de ingeniería	21
4 Resultados	26
Conclusiones	27
Recomendaciones	28
Apéndices	29
A Proyectos y Avals	30

B	Otro reporte	31
B.1	Otra sección de muestra	31
B.2	Otro ensamble Industrial	31

Índice de figuras

1.1	Pilares de la Metodología XP	5
1.2	Vista del Arco de Moncloa	7

1	Tarjeta CRC # 1	3
1.1	Caption de Prueba	7
2.1	Historia de usuario # 1	10
2.2	Historia de usuario # 2	11
2.3	Historia de usuario # 3	12
3.1	Feasible triples for highly variable Grid, MLMMH.	20
3.1	continued from previous page	21
3.2	Historia de usuario # 4	21
3.3	Tarjeta CRC # 2	22
3.4	Tarjeta CRC # 3	22
3.5	Estimación de esfuerzo por historia de usuario	22
3.6	Estimación de esfuerzo por historia de usuario	23
3.7	Plan de duración de las iteraciones	23
3.8	Tarea de ingeniería # 1	23
3.9	Tarea de desarrollo # 1	23
3.10	Tarea de desarrollo # 2	24
3.11	Tarea de ingeniería # 2	24
3.12	Prueba de aceptación # 1	24
3.13	Prueba de aceptación # 2	24

Lista de algoritmos

1	Estrategia de BellmanKalaba	14
---	---------------------------------------	----

Lista de códigos fuentes

2.1	Búsqueda de máximo	15
2.2	Ejemplo de código en Java	15
2.3	Ejemplo de código en PHP	16
2.4	Ejemplo de código en PHP	16
2.5	Ejemplo de código en Python	17
2.6	Ejemplo de código en htmlcssjs	17
2.7	Ejemplo de código en htmlcssjs	18

El desarrollo vertiginoso de las nuevas tecnologías, particularmente en las ramas de la informática y las telecomunicaciones, evidencia que es esta la era con mayor evolución científico-técnica de todos los tiempos. Este desarrollo tecnológico acelerado incide sustancialmente en el mundo referencial del ser humano, pues, si bien facilita la adquisición y almacenamiento de nuevos conocimientos, nuevos enfoques y/o perspectivas y acciones que ayer mismo parecían inaccesibles pero, de la misma manera, le están condicionando y obligando a adaptaciones y replanteamientos en todos los órdenes de su existencia.

En el campo del desarrollo de software también se constata un avance sin precedentes a nivel mundial, lo cual posibilita la construcción de aplicaciones y sistemas informáticos capaces de solventar la amplia amalgama de problemas teóricos y prácticos de la sociedad. Diversas áreas de la vida diaria se ven afectadas continuamente por la industria del software, tales como los aeropuertos, los bancos, las grandes empresas, las instituciones policiales y militares entre otras.

Los sistemas de gestión son una rama importante dentro de la industria del desarrollo de software puesto que resuelven un gran número de situaciones a través de la automatización de procesos, lo cual facilita el trabajo en oficinas, elimina el tiempo de respuesta por parte de los proveedores de servicios a la población, suprime engorrosos bloques de papeles, propicia el trabajo de los seres humanos con grandes volúmenes de información, el control y manipulación de estadísticas y una innumerable cantidad de beneficios y posibilidades más, que a la postre, repercuten de forma positiva en la economía y la calidad de vida de todos aquellos países que explotan este tipo de sistemas informáticos.

En Cuba, con la intención de utilizar la amplia gama de posibilidades que brinda la industria del software se trabaja hace ya algún tiempo en el desarrollo de este tipo de herramientas, donde la Universidad de las Ciencias Informáticas asume un rol protagónico en esta tarea. Justamente, la informatización de los procesos que se llevan a cabo en una empresa o entidad resulta un elemento clave. Un proceso que resulta prioritario automatizar por la importancia que tiene en la sociedad se encuentra relacionado con la gestión de citas pues de este dependerá el eficiente control y mejor planificación del tiempo en las oficinas de trámites.

La sociedad civil de servicios Transconsul [Sociedad Anónima \(S.A.\)](#), constituida en el mes de abril de 2020, es una organización que ofrece servicios legales a personas naturales y jurídicas en términos de

asesoría, asistencia, representación y legalización de documentos. Asimismo, posee competencia legal para contratar sus servicios tanto a clientes nacionales como extranjeros. Dicha sociedad mercantil enfrenta diversos desafíos en la gestión de citas y comunicación, lo que ha provocado una percepción negativa por parte de los clientes. Esto se puede percibir en la valoración de 1.8, así como los comentarios negativos en su página de Facebook, la poca interacción de los usuarios respecto a las publicaciones en X, además de la ausencia de respuestas por parte de la organización en Google. Los clientes expresan frustración debido a la dificultad para agendar citas pues actualmente estas solo se pueden programar mediante llamadas telefónicas en un horario específico (lunes de 9:00 AM a 4:00 PM). En primer lugar, los números de teléfono ofrecidos para dicha actividad se corresponden con líneas fijas, de modo que eso entorpece aún más la capacidad de la persona para realizar su solicitud. En segundo lugar, se advierte una brecha en las respuestas que Transconsul S.A ofrece a consultas y quejas por la lentitud en la gestión de los servicios, tanto a través de sus redes sociales como en las comunicaciones vía telefónica. Esta situación actual proyecta una imagen negativa de la organización y también limita la eficiencia de la misma.

A partir de la situación antes descrita se plantea como problema de la investigación: ¿Cómo mejorar la gestión de los servicios de Transconsul S.A.?

Teniendo como objeto de estudio: Sistemas de gestión de servicios.

Se define como objetivo general: Desarrollar una aplicación web para la mejora en la gestión de servicios de Transconsul S.A. que garantice la satisfacción del cliente y mejore la imagen pública de la empresa; enmarcado en el campo de acción: El proceso de gestión de servicios de Transconsul S.A.

Para dar cumplimiento al objetivo planteado se trazan las siguientes tareas de investigación:

- Análisis de los referentes teóricos sobre mejores prácticas en sistemas de gestión de citas en línea, servicios y comunicación con los clientes.
- Identificación de las necesidades y requisitos específicos de Transconsul S.A. para la gestión de citas y servicios legales.
- Desarrollo de una aplicación web que incluya un sistema de gestión de citas y servicios.
- Implementación de la solución tecnológica para asegurar la satisfacción del cliente y renovar la imagen pública de la empresa.
- Realización de pruebas funcionales y de usabilidad para validar el correcto funcionamiento y la aceptación de la aplicación web por parte de los usuarios.

Para el desarrollo de la investigación se emplean los siguientes métodos científicos:

Métodos teóricos:

- Analítico-Sintético: para el estudio de la situación actual y la identificación de requisitos y necesidades.
- Modelación: para el diseño de la aplicación y la conceptualización de sus componentes y funcionalidades.

Métodos empíricos:

- Entrevistas: con empleados y clientes de Transconsul S.A. para recoger sus necesidades y expectativas.
- Observación: análisis de la interacción actual de los clientes con Transconsul S.A..
- Análisis de Redes Sociales: Evaluación de la retroalimentación de los clientes en las redes sociales de Transconsul S.A., identificando patrones comunes en las quejas y sugerencias para mejorar los servicios de la empresa.

Tabla 1. Tarjeta CRC # 1

Tarjeta CRC	
Clase: GeneticAlgorithm	
Responsabilidad	Colaboración
<ul style="list-style-type: none">• Create gens• Create mutation• Create recombination	Population ProbabilityOperator

1.1. Introducción al capítulo

En este capítulo se sintetiza la búsqueda y análisis de la información relacionada con el dominio del problema. Son descritos los conceptos fundamentales relacionados con la investigación y analizadas algunas soluciones existentes referentes al mismo entorno. Aborda además, la selección y descripción de las principales tecnologías y herramientas que serán utilizadas para llevar a cabo la implementación de la solución a desarrollar.

1.2. Sistema de gestión de citas y servicios

1.3. Sistemas Homólogos

1.4. Metodología de desarrollo de software

Se siguieron los pasos que define la metodología [Programación Extrema \(XP\)](#) para guiar el proceso de desarrollo de software, atendiendo al esquema que se proponen en ([escribano2002introduccion](#)), adoptando sus 4 fases (Planificación, Diseño, Desarrollo y Prueba). [XP](#) es la más destacada de los procesos ágiles de desarrollo de software formulada por Knet Beck⁷. Tiene éxito porque hace hincapié en la satisfacción del cliente. Faculta a sus desarrolladores para responder con seguridad a las necesidades cambiantes de los clientes, incluso en etapas tardías del ciclo de vida del producto. Enfatiza el trabajo en equipo. Los jefes de proyecto, clientes y desarrolladores son socios iguales en un equipo de colaboración que se auto-organiza en torno al problema a resolver de la forma más eficiente posible. [XP](#) mejora un proyecto de software en cinco aspectos esenciales; la comunicación, la sencillez, la retroalimentación, el respeto y el valor. Los programadores extremos se comunican constantemente con sus clientes y colegas de trabajo, mantienen su diseño simple y limpio. Reciben retroalimentación probando su software a partir del primer día y entregan el sistema a los clientes tan pronto como sea posible e implementan cambios como se sugiere. Cada pequeño

éxito profundiza su respeto por las contribuciones únicas de cada uno y cada miembro del equipo. Con esta base los programadores extremos son capaces de responder con valentía a las cambiantes necesidades y la tecnología (jaskowicz2008reglas).

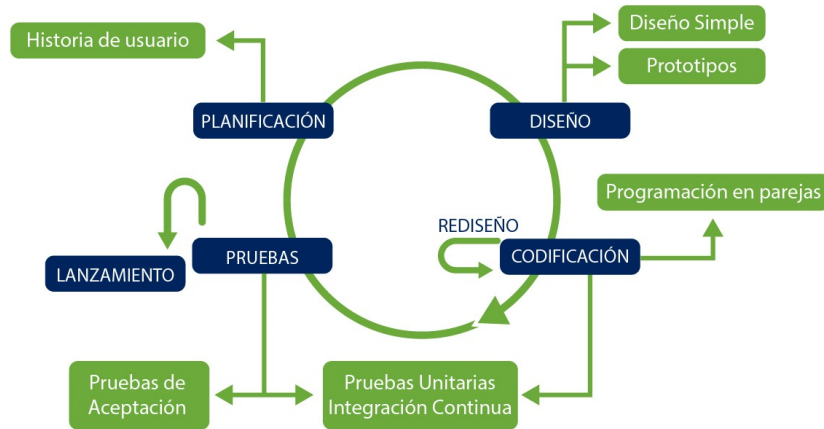


Figura 1.1. Pilares de la Metodología XP

1.5. Herramientas y tecnología de desarrollo de software

La elección y uso de herramientas y tecnologías apropiadas es crucial para el éxito del desarrollo del sistema de gestión de citas y servicios de Transconsul [XP](#). Al evaluar y seleccionar cuidadosamente estas herramientas, los programadores pueden trabajar de manera más eficiente y efectiva, lo que a su vez se traducirá en un producto final de alta calidad. Por lo tanto, es importante llevar a cabo una investigación exhaustiva de las herramientas y tecnologías disponibles, con el fin de identificar aquellas que mejor se adapten a las necesidades específicas del proyecto. De esta manera, se puede garantizar que el equipo de desarrollo tenga acceso a las herramientas y tecnologías necesarias para trabajar de manera efectiva y lograr los objetivos del proyecto de manera satisfactoria.

- Marco de trabajo: Con Django, puedes llevar aplicaciones web desde el concepto hasta el lanzamiento en cuestión de horas. Django se encarga de gran parte de las molestias del desarrollo web, por lo que puedes concentrarte en escribir tu aplicación sin necesidad de reinventar la rueda. Es gratuito y de código abierto.
 - Completamente cargado: Django incluye docenas de extras que puedes usar para manejar tareas comunes de desarrollo web. Django se encarga de la autenticación de usuarios, la administración de contenidos, los mapas del sitio, las fuentes RSS y muchas más tareas, desde el primer momento.

- Tranquilamente seguro: Django se toma en serio la seguridad y ayuda a los desarrolladores a evitar muchos errores de seguridad comunes, como la inyección SQL, secuencias de comandos entre sitios, falsificación de solicitudes entre sitios y clickjacking. Su sistema de autenticación de usuarios proporciona una forma segura de administrar cuentas de usuarios y contraseñas.
 - Extremadamente escalable: Algunos de los sitios más concurridos del planeta utilizan la capacidad de Django para escalar de forma rápida y flexible para satisfacer las demandas de tráfico más intensas.
 - Increíblemente versátil: Empresas, organizaciones y gobiernos han utilizado Django para construir todo tipo de cosas, desde sistemas de gestión de contenidos hasta redes sociales y plataformas informáticas científicas.
- Entorno de desarrollo: Visual Studio Code es un editor de código fuente liviano pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux. Viene con soporte integrado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes y tiempos de ejecución (como C++, Java, Python, PHP, Go, .NET)
- Un 'entorno' en Python es el contexto en el que se ejecuta un programa Python que consta de un intérprete y cualquier cantidad de paquetes instalados.

El venv módulo admite la creación de "entornos virtuales" livianos, cada uno con su propio conjunto independiente de paquetes Python instalados en sus sitedirectorios. Un entorno virtual se crea sobre una instalación de Python existente, conocida como Python ?base? del entorno virtual, y opcionalmente puede aislarse de los paquetes en el entorno base, de modo que solo estén disponibles aquellos instalados explícitamente en el entorno virtual.

Cuando se utilizan desde un entorno virtual, las herramientas de instalación comunes, como pip , instalarán paquetes de Python en un entorno virtual sin necesidad de que se le indique explícitamente que lo haga.

- Sistema Gestor de Base de Datos: se utilizó MySQL, este es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. Es uno de los gestores más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación, es debida a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración ([daniel_pecos_martinez_postgresql_2024](#)). Las características fundamentales que refleja para su elección por encima de otros gestores de base de datos son: el coste gratuito, la velocidad operacional, facilidad de uso y de integración con la mayor parte de los entornos de programación, la existencia de una nutrida y activa comunidad ([gilfillan_biblia_2003](#)).

Escribir aquí el capítulo 1. Ejemplo de acrónimo es la [Universidad de las Ciencias Informáticas \(UCI\)](#). Esto es una cita de ejemplo ([Ou2013](#)). En la próxima oración se muestra un ejemplo de un acrónimo en

inglés. Los sistemas de [Supervisión, Control y Adquisición de Datos \(SCADA, por sus siglas en inglés\)](#) son de gran importancia para la industria.

1.6. Sección de prueba de la UCI

Esto es una utilización de una palabra del glosario. El término [engine](#) es utilizado en este ejemplo. Esta es una prueba de número: 5.3 donde se utiliza la coma automáticamente como separador.

Esto es una cita de ejemplo ([Alfadhilani2011](#); [Mathew2010](#)).



Figura 1.2. Vista del Arco de Moncloa

En la Tabla [1.1](#) se observa ...

Tabla 1.1. Caption de Prueba

No. 1	No. 2	No. 3	No. 4	No. 5
12	12	56	56	56
12	12	56	56	56

- Uno
- Dos
- Tres
- Cuatro

- Cinco

2.1. Introducción al capítulo

En el presente capítulo se presenta una propuesta de solución para los problemas detectados en el estudio del estado del arte realizado en el Capítulo 1. Se mencionan los roles que intervienen en la solución. También se definen las HU, la planificación de entrega de versiones del producto y el diseño del sistema de acuerdo a las fases que propone XP.

2.2. Propuesta de solución

Luego de haber analizado las necesidades del sistema y seleccionado las herramientas para la implementación, se definen los módulos a desarrollar para dar solución al problema planteado.

- Autenticación y Autorización: Este módulo se encargaría de gestionar la autenticación de usuarios, el registro de cuentas y la autorización de acceso a diferentes partes de la aplicación.
- Gestión de Citas: Este módulo sería el núcleo de la aplicación y se encargaría de permitir a los usuarios programar, modificar y cancelar citas. Debería incluir un calendario interactivo, notificaciones de citas programadas y recordatorios automáticos.
- Gestión de Clientes: Aquí se manejaría toda la información relacionada con los clientes, incluyendo su perfil, historial de citas, documentos legalizados y cualquier otra información relevante.
- Servicios Legales y Legalización: Este módulo se enfocaría en la gestión de los servicios legales ofrecidos por el bufete de abogados, así como en el proceso de legalización de documentos. Debería permitir a los usuarios solicitar servicios específicos, enviar documentos y realizar seguimiento del estado de sus trámites.
- Comunicación y Notificaciones: Este módulo sería responsable de facilitar la comunicación entre el bufete de abogados y sus clientes, a través de mensajes directos, correos electrónicos o notificaciones automáticas sobre el estado de las citas y servicios solicitados.

- **Administración del Sistema:** Este módulo sería utilizado por los administradores del sistema para gestionar usuarios, configurar ajustes de la aplicación, generar informes y realizar tareas de mantenimiento.
- **Análisis y Reportes:** Este módulo se encargaría de recopilar datos sobre el uso de la aplicación, el rendimiento del sistema y la satisfacción del cliente, para generar informes y análisis que ayuden a mejorar la eficiencia y calidad del servicio.

Para el desarrollo de los módulos propuestos se siguen los pasos que establece la metodología ágil XP, donde se respetará el esquema que explica (**escribano2002introduccion**).

2.3. Fase I: Planificación

Es la fase en la que se define el alcance general del proyecto. En esta el cliente define lo que necesita mediante la redacción de HU y establece la prioridad de cada una. Luego, los programadores estiman los tiempos de desarrollo en base a esta información. Las estimaciones realizadas en esta fase son primarias, debido a que estarán basadas en datos de muy alto nivel y podrían variar cuando se analicen en cada iteración. Además se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses (**escribano2002introduccion; joskowicz2008reglas**).

Las HU son la técnica que utiliza XP para especificar los requisitos de software, estas deben ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a las tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra HU. Las estimaciones de esfuerzo, asociado a la implementación de las historias, la establecen los programadores, utilizando como medida el punto. Un punto, equivale a una semana ideal de programación (5 días laborables). Las historias, generalmente, valen de 1 a 3 puntos. (**escribano2002introduccion**). A continuación se describen las HU definidas para llevar a cabo el desarrollo de los módulos.

Tabla 2.1. Historia de usuario # 1

Historia de usuario	
Número: 1	Nombre: Registro de Usuario
Usuario: Administrador, Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Téc. Carlos Brayan Rámila Chorens	

Continúa en la próxima página

Tabla 2.1. Continuación de la página anterior

<p>Descripción: Como usuario nuevo, quiero poder registrarme en la aplicación proporcionando mi información personal, como nombre, dirección de correo electrónico y contraseña, para acceder a las funcionalidades de la aplicación.</p> <ul style="list-style-type: none"> • nombre • dirección de correo electrónico • contraseña • carnet de identidad
<p>Observaciones:</p> <ul style="list-style-type: none"> • La aplicación debe permitir al usuario completar un formulario de registro con campos para nombre, correo electrónico y contraseña. • Los campos del formulario deben validar que la información ingresada sea válida y esté en el formato correcto. • Después de enviar el formulario, la aplicación debe crear una cuenta de usuario y almacenarla en la base de datos. • Se debe enviar un correo electrónico de confirmación al usuario registrado para verificar su dirección de correo electrónico. • El usuario debe poder iniciar sesión después de completar el registro exitosamente.

Tabla 2.2. Historia de usuario # 2

Historia de usuario	
Número: 2	Nombre: Iniciar Sesión
Usuario: Administrador, Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Téc. Carlos Brayan Rámila Chorens	
<p>Descripción: Como usuario registrado, quiero poder iniciar sesión en la aplicación utilizando mi dirección de correo electrónico y contraseña, para acceder a mis datos y realizar acciones dentro de la aplicación.</p> <ul style="list-style-type: none"> • dirección de correo electrónico • contraseña 	

Continúa en la próxima página

Tabla 2.2. Continuación de la página anterior

Observaciones:

- La aplicación debe proporcionar un formulario de inicio de sesión con campos para correo electrónico y contraseña.
- Los campos del formulario deben validar que la información ingresada sea válida y coincida con los datos almacenados en la base de datos.
- Después de enviar el formulario, la aplicación debe autenticar al usuario y redirigirlo a la página principal si las credenciales son válidas.
- Si las credenciales son inválidas, la aplicación debe mostrar un mensaje de error al usuario indicando que las credenciales son incorrectas.
- La sesión del usuario debe mantenerse activa mientras navega por la aplicación, permitiéndole acceder a las funcionalidades protegidas sin necesidad de volver a iniciar sesión.

Tabla 2.3. Historia de usuario # 3

Historia de usuario	
Número: 3	Nombre: Gestión de Perfiles de Usuario
Usuario: Administrador	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Téc. Carlos Brayan Rámila Chorens	
Descripción: Como administrador, quiero poder gestionar los perfiles de usuario, incluyendo la capacidad de crear, editar y eliminar cuentas de usuario, para mantener el control sobre quién tiene acceso a la aplicación. <ul style="list-style-type: none"> • nombre • dirección de correo electrónico • contraseña • carnet de identidad 	

Continúa en la próxima página

Tabla 2.3. Continuación de la página anterior

Observaciones:

- La aplicación debe proporcionar una interfaz de administración donde el administrador pueda ver una lista de todos los usuarios registrados.
- El administrador debe poder crear nuevos perfiles de usuario, proporcionando información como nombre, dirección de correo electrónico y contraseña.
- Se debe implementar la funcionalidad de edición para permitir al administrador actualizar la información de los perfiles de usuario existentes.
- El administrador debe poder eliminar cuentas de usuario cuando sea necesario, lo que resultará en la eliminación permanente de los datos asociados con esa cuenta.
- Se deben implementar controles de acceso adecuados para garantizar que solo el administrador tenga acceso a estas funcionalidades de gestión de perfiles.

2.4. Sección de prueba

Escribir aquí el capítulo 2. Prueba de glosario [engine](#). A continuación un ejemplo de enlace al anexo [Bloque Corona](#). Este es un ejemplo de matriz con bordes numerados ([Mathew2010](#)).

$$\begin{array}{cc}
 \begin{array}{c} \text{Padre} \end{array} & \begin{array}{c} 1 \quad 2 \quad 3 \quad 4 \\ \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \begin{array}{l} \rightarrow \text{Seleccionada aleatoriamente} \\ \rightarrow \text{Seleccionada aleatoriamente} \end{array}
 \end{array}
 \quad
 \begin{array}{cc}
 \begin{array}{c} \text{Madre} \end{array} & \begin{array}{c} 1 \quad 2 \quad 3 \quad 4 \\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}
 \end{array}$$

$$\begin{array}{cc}
 & \begin{array}{c} \text{Hijo} \end{array} \\
 & \begin{array}{c} 1 \quad 2 \quad 3 \quad 4 \\ \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}
 \end{array}
 \quad (2.4.1)$$

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in,

pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Este es un ejemplo de cita (**Mathew2010**) Esta es una prueba de cómo se escribe un pseudocódigo en \LaTeX .

Algoritmo 1 Estrategia de BellmanKalaba

```
1: procedure BELLMANKALABA( $G, u, l, p$ )
2:   for all  $v \in V(G)$  do
3:      $l(v) \leftarrow \infty$ 
4:   end for
5:    $l(u) \leftarrow 0$ 
6:   repeat
7:     for  $i \leftarrow 1, n$  do
8:        $min \leftarrow l(v_i)$ 
9:       for  $j \leftarrow 1, n$  do
10:        if  $min > e(v_i, v_j) + l(v_j)$  then
11:           $min \leftarrow e(v_i, v_j) + l(v_j)$ 
12:           $p(i) \leftarrow v_j$ 
13:        end if
14:      end for
15:       $l'(i) \leftarrow min$ 
16:    end for
17:     $changed \leftarrow l \neq l'$ 
18:     $l \leftarrow l'$ 
19:  until  $\neg changed$ 
20: end procedure
```

2.5. Ejemplos de código fuente

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat

ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Este es un ejemplo de cita (**Mathew2010**)

Código fuente 2.1. Búsqueda de máximo

```
int maxSearch( const int* v)
{
    /**
     * This is a text that takes
     * two lines
     */
    int currentMax = 0;
    for(int i = 0; i < MAX_NUMBER; i++)
    {
        currentMax = max(v[i], currentMax); // here we are using std::max
    }

    return currentMax;
}
```

Código fuente 2.2. Ejemplo de código en Java

```
1 /**
2  * HelloWorldApp class prints
3  * "Hello World!" to standard output
4  */
5 public class HelloWorldApp{
6     public static void main(String argv[])
```

```
7 {
8     // Display string
9     System.out.println("Hello World!");
10 }
11 }
```

Código fuente 2.3. Ejemplo de código en PHP

```
1 <?php
2 /**
3  * Retrieves a list of models based on the current search/filter conditions.
4  * @return CActiveDataProvider the data provider that can return the models
5  * based on the search/filter conditions.
6  */
7 public function search()
8 {
9     // Warning: Please modify the following code to remove attributes that
10    // should not be searched.
11
12    $criteria=new CDbCriteria;
13    $criteria->compare('idtarea',$this->idtarea);
14    $criteria->compare('nombre',$this->nombre,true);
15    $criteria->compare('descripcion',$this->descripcion,true);
16    $criteria->compare('fecha',$this->fecha,true);
17    $criteria->compare('idinstructora',$this->idinstructora);
18
19    // $user=Yii::app()->user->name;
20    $rol=User::model()->getRolThisUser($user);
21    $rol=User::model()->getNameRol($rol);
22    if('Instructora'==$rol)
23    {
24        $criteria->join="JOIN instructora ON(t.idinstructora=instructora.
25        idinstructora ) ";
26        $criteria->condition="instructora.idusuario='".$user."'";
27    }
28
29    return new CActiveDataProvider($this, array(
30        'criteria'=>$criteria,
31        'pagination'=>array('pageSize'=>5)
32    ));
33 }
```

Código fuente 2.4. Ejemplo de código en PHP

```
1 public function evidenciaXUsuarioAction($id)
2 {
3     $em = $this->getDoctrine()->getManager();
4
5     $usuario = $em->getRepository('GIVIPBundle:Usuario')->find($id);
6     if(!$usuario){
7         $this->get('session')->getFlashBag()->add('error', 'El usuario
8         seleccionado no existe en el sistema.');
```

```
10     }
11     $nombre = $usuario->getNombre() ." ". $usuario->getApellidos();
12     if($this->get('security.context')->isGranted('ROLE_VIP1')){
13         $consulta = $em->createQuery("select e, i, u from GIVIPBundle:Evidencia e
14                                     JOIN e.indicador i JOIN e.usuario u WHERE e.
15                                     usuario=:id");
16     }
17     else
18     {
19         $consulta = $em->createQuery("select e, i, u from GIVIPBundle:Evidencia e
20                                     JOIN e.indicador i JOIN e.usuario u WHERE e.usuario=:id and e.estado=
21                                     true");
22     }
23
24     $consulta->setParameter('id', $id);
25
26     $res = $consulta->getResult();
27     $paginator = $this->get('ideup.simple_paginator');
28     $paginator->setItemsPerPage(10, 'resultado');
29     $entities = $paginator->paginate($res, 'resultado')->getResult();
30
31     $deleteForm = $this->createDeleteForm(0);
32     $buscarForm = $this->createSearchForm();
33
34     return $this->render('GIVIPBundle:Evidencia:evidenciausuario.html.twig',
35         array(
36             'entities' => $entities, 'delete_form' => $deleteForm->createView(),
37             'buscar_form' => $buscarForm->createView(), 'usuario'=>$nombre
38         ));
39 }
```

Código fuente 2.5. Ejemplo de código en Python

```
1 class TelegramRequestHandler(object):
2     def handle(self):
3         addr = self.client_address[0]          # Client IP-adress
4         telegram = self.request.recv(1024)     # Recieve telegram
5         print "From: %s, Received: %s" % (addr, telegram)
6         return
```

Código fuente 2.6. Ejemplo de código en htmlcssjs

```
1 @media only screen and (min-width: 768px) and (max-width: 991px) {
2
3     #main {
4         width: 712px;
5         padding: 100px 28px 120px;
6     }
7
8     /* .mono {
9         font-size: 90%;
10    } */
11
12    .cssbtn a {
13        margin-top: 10px;
```

```
14     margin-bottom: 10px;
15     width: 60px;
16     height: 60px;
17     font-size: 28px;
18     line-height: 62px;
19 }
```

Código fuente 2.7. Ejemplo de código en htmlcssjs

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Listings Style Test</title>
5     <meta charset="UTF-8">
6     <style>
7       /* CSS Test */
8       * {
9         padding: 0;
10        border: 0;
11        margin: 0;
12      }
13    </style>
14    <link rel="stylesheet" href="css/style.css" />
15  </head>
16  <header> hey </header>
17  <article> this is a article </article>
18  <body>
19    <!-- Paragraphs are fine -->
20    <div id="box">
21      <p>
22        Hello World
23      </p>
24      <p>Hello World</p>
25      <p id="test">Hello World</p>
26      <p></p>
27    </div>
28    <div>Test</div>
29    <!-- HTML script is not consistent -->
30    <script src="js/benchmark.js"></script>
31    <script>
32      function createSquare(x, y) {
33        // This is a comment.
34        var square = document.createElement('div');
35        square.style.width = square.style.height = '50px';
36        square.style.backgroundColor = 'blue';
37
38        /*
39         * This is another comment.
40         */
41        square.style.position = 'absolute';
42        square.style.left = x + 'px';
43        square.style.top = y + 'px';
44
45        var body = document.getElementsByTagName('body')[0];
46        body.appendChild(square);
47      };
```

```
48
49     // Please take a look at +=
50     window.addEventListener('mousedown', function(event) {
51         // German umlaut test: Berührungspunkt ermitteln
52         var x = event.touches[0].pageX;
53         var y = event.touches[0].pageY;
54         var lookAtThis += 1;
55     });
56 </script>
57 </body>
58 </html>
```

Ejemplo de tabla grande

3.1. Tabla aleatoria

Tabla 3.1. Feasible triples for highly variable Grid, MLMMH.

Time (s)	Triple chosen	Other feasible triples
0	(1, 11, 13725)	(1, 12, 10980), (1, 13, 8235), (2, 2, 0), (3, 1, 0)
2745	(1, 12, 10980)	(1, 13, 8235), (2, 2, 0), (2, 3, 0), (3, 1, 0)
5490	(1, 12, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
8235	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
10980	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
13725	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
16470	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
19215	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
21960	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
24705	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
27450	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
30195	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)
32940	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
35685	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
38430	(1, 13, 10980)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
41175	(1, 12, 13725)	(1, 13, 10980), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
43920	(1, 13, 10980)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
46665	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)
49410	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)
52155	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
54900	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
57645	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
60390	(1, 12, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
63135	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
65880	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
68625	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)
71370	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
74115	(1, 12, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
76860	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
79605	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)

Continued on next page

Tabla 3.1. continued from previous page

Time (s)	Triple chosen	Other feasible triples
82350	(1, 12, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
85095	(1, 12, 13725)	(1, 13, 10980), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
87840	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
90585	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
93330	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
96075	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
98820	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
101565	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
104310	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
107055	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
109800	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
112545	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
115290	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
118035	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
120780	(1, 13, 16470)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
123525	(1, 13, 13725)	(2, 2, 2745), (2, 3, 0), (3, 1, 0)
126270	(1, 12, 16470)	(1, 13, 13725), (2, 2, 2745), (2, 3, 0), (3, 1, 0)
129015	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)
131760	(2, 2, 2745)	(2, 3, 0), (3, 1, 0)

3.2. Tablas de ingeniería

Tabla 3.2. Historia de usuario # 4

Historia de usuario	
Número: 4	Nombre: Cargar archivo
Usuario: Especialista	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 0.8	Iteración asignada: 1
Programador responsable: Ernesto Gil Capote	
<p>Descripción: Permite cargar los datos de los casos de estudio a utilizar en el módulo con el objetivo de determinar secuencias factibles. Estos están almacenados en archivos con extensión .asp. La estructura está compuesta por los parámetros de entrada de la aplicación para cada caso de estudio tales como: dimensión, nombre de la estructura mecánica, cantidad de piezas, la MD correspondiente y el vector de herramientas en caso de que lo posea.</p> <ul style="list-style-type: none"> • rojo • blanco 	
<p>Observaciones: En caso de que el usuario no cargue un archivo, el sistema debe mostrar un mensaje de alerta con dicha explicación. Si se introduce un archivo que contiene caracteres no válidos o falta alguno de los elementos necesarios para la ejecución del módulo, este debe lanzar la excepción correspondiente. En caso de que el usuario no cargue un archivo, el sistema debe mostrar un mensaje de alerta con dicha explicación. Si se introduce un archivo que contiene caracteres no válidos o falta alguno de los elementos necesarios para la ejecución del módulo, este debe lanzar la excepción correspondiente.</p>	

Esta es referencia a una tabla de historia de usuario [3.2](#)

Tabla 3.3. Tarjeta CRC # 2

Tarjeta CRC	
Clase: AlgoritmoGenetico	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> • Crear el conjunto de genes necesarios para la creación de un individuo • Realizar el proceso de mutación y recombinación • Crear el conjunto de genes necesarios para la creación de un individuo • Realizar el proceso de mutación y recombinación 	Población OperadorProbabilidad OperadorParejas OperadorReproduccion Población

Tabla 3.4. Tarjeta CRC # 3

Tarjeta CRC	
Clase: AlgoritmoGenetico	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> • Crear el conjunto de genes necesarios para la creación de un individuo • Realizar el proceso de mutación y recombinación • Crear el conjunto de genes necesarios para la creación de un individuo • Realizar el proceso de mutación y recombinación 	Población OperadorProbabilidad OperadorParejas OperadorReproduccion Población

Tabla 3.5. Estimación de esfuerzo por historia de usuario

Iteración	Historias de usuario		Puntos estimados (semanas)
1	1	Cargar Archivo	0.8
	2	Determinar Secuencia de Ensamble	0.2
	3	Gestionar restricción cambios de herramienta	0.5
	4	Graficar secuencias de ensamble	0.6
2	5	Graficar secuencias de ensamble	0.3
	6	Graficar secuencias de ensamble	0.3
	7	Graficar secuencias de ensamble	0.3
	8	Graficar secuencias de ensamble	0.2
	9	Graficar secuencias de ensamble	0.2
3	10	Determinar Secuencia de Ensamble	0.5
4	11	Graficar secuencias de ensamble	0.1
	12	Graficar ensamble	0.1
Total			4.1

Tabla 3.6. Estimación de esfuerzo por historia de usuario

Iteración	Historias de usuario		Puntos estimados (semanas)
1	1	Cargar Fichero	0.8
	2	Determinar Secuencia de Ensamble	0.2
	3	Gestionar restricción cambios de herramienta	0.5
	4	Esta es la prueba para el tinguiri	0.5
2	5	Graficar secuencias de ensamble	0.3
	6	Graficar secuencias de ensamble	0.3
	7	Graficar secuencias de ensamble	0.2
	8	Graficar secuencias de ensamble	0.2
3	9	Determinar Secuencia de Ensamble	2.5
4	10	Graficar secuencias de ensamble	1.1
Total			6.6

Esto es una prueba de Tabla 3.5

Tabla 3.7. Plan de duración de las iteraciones

Iteración	Historias de usuario		Duración (semanas)
1	1	Cargar Fichero	3
	2	Determinar Secuencia de Ensamble	
	3	Gestionar restricción cambios de herramienta	
	4	Esta es la prueba para el tinguiri	
2	5	Graficar secuencias de ensamble	1.0
	6	Graficar secuencias de ensamble	
	7	Graficar secuencias de ensamble	
	8	Graficar secuencias de ensamble	
3	9	Determinar Secuencia de Ensamble	2.5
4	10	Graficar secuencias de ensamble	1.1
Total			7.6

Esto es una prueba de Tabla 3.7

Tabla 3.8. Tarea de ingeniería # 1

Tarea	
Número de tarea: 1	Número de Historia de usuario: 4
Nombre de la tarea: Registrar usuario	
Tipo de tarea: Registrarse	Puntos estimados: 1
Fecha de inicio: 6 de mayo de 2014	Fecha de fin: 24 de octubre de 2014
Descripción: El usuario introduce sus datos para poder registrarse.	

Tabla 3.9. Tarea de desarrollo # 1

Tarea	
Número de tarea: 1	Número de Historia de usuario: 4
Nombre de la tarea: Registrar usuario	
Tipo de tarea: Registrarse	Puntos estimados: 1
Fecha de inicio: 6 de mayo de 2014	Fecha de fin: 24 de octubre de 2014

Continúa en la próxima página

Tabla 3.9. Continuación de la página anterior

Programador responsable: Alberto Medina Ramírez
Descripción: El usuario introduce sus datos para poder registrarse.

Tabla 3.10. Tarea de desarrollo # 2

Tarea	
Número de tarea: 2	Número de Historia de usuario: 4
Nombre de la tarea: Registrar usuario	
Tipo de tarea: Registrarse	Puntos estimados: 1
Fecha de inicio: 6 de mayo de 2014	Fecha de fin: 24 de octubre de 2014
Descripción: El usuario introduce sus datos para poder registrarse.	

Tabla 3.11. Tarea de ingeniería # 2

Tarea	
Número de tarea: 2	Número de Historia de usuario: 4
Nombre de la tarea: Registrar usuario	
Tipo de tarea: Registrarse	Puntos estimados: 1
Fecha de inicio: 6 de enero de 2014	Fecha de fin: 24 de noviembre de 2014
Programador responsable: Alberto Medina Ramírez	
Descripción: El usuario introduce sus datos para poder registrarse.	

Referencia [3.9](#)

Tabla 3.12. Prueba de aceptación # 1

Caso de prueba de aceptación	
Código: HU2_P1	Historia de usuario: 2
Nombre: Autenticar usuario en el sistema.	
Descripción: Prueba para la funcionalidad autenticar usuario.	
Condiciones de ejecución: El usuario debe estar previamente registrado. El usuario y contraseña deben ser válidos.	
Pasos de ejecución: Se intenta autenticar un usuario en el sistema con los datos válidos.	
Resultados esperados: El usuario se autentifica correctamente en el sistema.	

Tabla 3.13. Prueba de aceptación # 2

Caso de prueba de aceptación	
Código: HU2_P1	Historia de usuario: 2
Nombre: Autenticar usuario en el sistema.	
Descripción: Prueba para la funcionalidad autenticar usuario.	
Condiciones de ejecución: El usuario debe estar previamente registrado. El usuario y contraseña deben ser válidos.	

Continúa en la próxima página

Tabla 3.13. Continuación de la página anterior

Pasos de ejecución: Se intenta autenticar un usuario en el sistema con los datos válidos.
Resultados esperados: El usuario se autentifica correctamente en el sistema.

Prueba de link [3.12](#)

Resultados

Escribir aquí el capítulo 3. Otra prueba más [engine](#). Esto es un ejemplo de cómo se puede *linkear* el anexo [Controlador Industrial](#).

Esto es otra cita de la inicial como ejemplo también (**Ou2013**).

Otra prueba más [engine](#)

Otra prueba más [engine](#)

Esto es una prueba de link al Anexo [B.2](#). Esto es una prueba de link a la palabra [sentencia](#)

Conclusiones

Escribir aquí las conclusiones.

Recomendaciones

Escribe aquí las recomendaciones de tu trabajo.

Apéndices

Proyectos y Avaes



REPORTE DEL PLANIFICADOR DE SECUENCIA DE ENSAMBLE

Ing. Maybel Díaz Capote
Universidad de las Ciencias Informáticas, La Habana, Cuba
Fecha: sábado 5 de octubre de 2013

DETALLES DEL REPORTE:

Caso de estudio: Bloque Corona
Reorientaciones: sí
Cambios de herramientas: no
Vértices: 17
Aristas: 136
Número total de hormigas: 11

Secuencias de ensamble:

Hormiga: 1
Reorientaciones: 2
Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(1, +z)
Hormiga: 2
Reorientaciones: 3
Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(7, +y)-(17, +y)-(1, +y)
Hormiga: 3
Reorientaciones: 3
Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(3, +z)-(6, +z)-(17, +z)-(1, +z)-(13, +z)-(7, +y)-(4, +y)-(2, +y)
Hormiga: 4
Reorientaciones: 2
Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(13, +z)-(1, +z)-(6, +z)-(3, +z)-(2, +z)
Hormiga: 5
Reorientaciones: 3
Salida:
(11, +z)-(10, +z)-(16, +x)-(14, +x)-(15, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(1, +z)-(2, +z)-(13, +z)-(8, +x)-(3, +x)
Hormiga: 6
Reorientaciones: 2
Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(1, +z)-(6, +z)-(2, +z)-(13, +z)-(3, +z)

B.1. Otra sección de muestra

De una sola sentencia

B.2. Otro ensamble Industrial



REPORTE DEL PLANIFICADOR DE SECUENCIA DE ENSAMBLE

Ing. Maybel Díaz Capote
Universidad de las Ciencias Informáticas, La Habana, Cuba
Fecha: sábado 5 de octubre de 2013

DETALLES DEL REPORTE:

Caso de estudio: Bloque Corona
Reorientaciones: sí
Cambios de herramientas: no
Vértices: 17
Aristas: 136
Número total de hormigas: 11

Secuencias de ensamble:

Hormiga: 1
Reorientaciones: 2

Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(1, +z)

Hormiga: 2
Reorientaciones: 3

Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(7, +z)-(17, +z)-(1, +z)

Hormiga: 3
Reorientaciones: 3

Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(3, +z)-(6, +z)-(17, +z)-(1, +z)-(13, +z)-(7, +z)-(4, +z)-(2, +z)

Hormiga: 4
Reorientaciones: 2

Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(13, +z)-(1, +z)-(6, +z)-(3, +z)-(2, +z)

Hormiga: 5
Reorientaciones: 3

Salida:
(11, +z)-(10, +z)-(16, +x)-(14, +x)-(15, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(1, +z)-(2, +z)-(13, +z)-(8, +x)-(3, +x)

Hormiga: 6
Reorientaciones: 2

Salida:
(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(1, +z)-(6, +z)-(2, +z)-(13, +z)-(3, +z)



REPORTE DEL PLANIFICADOR DE SECUENCIA DE ENSAMBLE

Ing. Maybel Díaz Capote
Universidad de las Ciencias Informáticas, La Habana, Cuba
Fecha: sábado 5 de octubre de 2013

DETALLES DEL REPORTE:

Caso de estudio: Bloque Corona
Reorientaciones: sí
Cambios de herramientas: no
Vértices: 17
Aristas: 136
Número total de hormigas: 11

Secuencias de ensamble:

Hormiga: 1

Reorientaciones: 2

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(1, +z)

Hormiga: 2

Reorientaciones: 3

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(7, -y)-(17, -y)-(1, -y)

Hormiga: 3

Reorientaciones: 3

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(3, +z)-(6, +z)-(17, +z)-(1, +z)-(13, +z)-(7, +y)-(4, +y)-(2, +y)

Hormiga: 4

Reorientaciones: 2

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(13, +z)-(1, +z)-(6, +z)-(3, +z)-(2, +z)

Hormiga: 5

Reorientaciones: 3

Salida:

(11, +z)-(10, +z)-(16, +x)-(14, +x)-(15, +x)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(1, +z)-(2, +z)-(13, +z)-(8, +x)-(3, +x)

Hormiga: 6

Reorientaciones: 2

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(1, +z)-(6, +z)-(2, +z)-(13, +z)-(3, +z)

Hormiga: 7

Reorientaciones: 2

Salida:

(8, -x)-(14, -x)-(15, -x)-(16, -x)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(2, +z)-(13, +z)-(1, +z)-(9, -z)-(10, -z)-(11, -z)-(12, -z)-(3, -z)

Hormiga: 8

Reorientaciones: 2

Salida:

(8, -x)-(14, -x)-(15, -x)-(16, -x)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(2, +z)-(13, +z)-(1, +z)-(3, -z)-(9, -z)-(10, -z)-(12, -z)-(11, -z)

Hormiga: 9

Reorientaciones: 3

Salida:

(8, -x)-(14, -x)-(15, -x)-(16, -x)-(10, -z)-(11, -z)-(9, +z)-(5, +z)-(4, +z)-(7, +z)-(17, +z)-(1, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)-(12, -x)

Hormiga: 10

Reorientaciones: 2

Salida:

(8, -x)-(14, -x)-(15, -x)-(16, -x)-(5, +z)-(4, +z)-(6, +z)-(7, +z)-(17, +z)-(2, +z)-(13, +z)-(1, +z)-(3, -z)-(9, -z)-(10, -z)-(11, -z)-(12, -z)

Hormiga: 11

Reorientaciones: 2

Salida:

(11, +z)-(10, +z)-(16, +x)-(15, +x)-(14, +x)-(8, +x)-(12, +z)-(9, +z)-(5, +z)-(7, +z)-(17, +z)-(4, +z)-(1, +z)-(6, +z)-(3, +z)-(2, +z)-(13, +z)

Secuencia de salida premiada**Reorientaciones: 2****Salida:**

(8, -x)-(15, -x)-(14, -x)-(16, -x)-(10, -z)-(11, -z)-(12, +z)-(9, +z)-(5, +z)-(4, +z)-(7, +z)-(6, +z)-(13, +z)-(3, +z)-(17, +z)-(2, +z)-(1, +z)