



Sistema informático para la distribución de uniforme escolar. Caso de estudio: provincia de Granma, Cuba

Computer system the distribution of school uniforms. Case study: province of Granma, Cuba

Yamira Medel Viltres

Universidad de Granma, Granma, Cuba

ymedelv@udg.co.cu

ORCID: 0000-0002-3485-7550

Fidel Enrique Castro Dieguez

Universidad de Granma, Granma, Cuba

fcastrod@udg.co.cu

ORCID: 0000-0002-5313-0787

Angel Enrique Figueredo León

Universidad de Granma, Granma, Cuba

afigueredol@udg.co.cu

ORCID: 0000-0002-9536-2826

Alberto Rubén Leyva Polo

Universidad de Granma, Granma, Cuba

alberpolo@nauta.cu

ORCID: 0000-0003-1133-9096

Adrián Almaguel Guerra

Universidad de Granma, Granma, Cuba

aalmaguelg@udg.co.cu

ORCID: 0000-0002-2963-677X

doi: <https://doi.org/10.36825/RITI.09.19.001>

Recibido: Abril 26, 2021

Aceptado: Agosto 9, 2021

Resumen: La distribución de uniforme escolar en la provincia de Granma es un proceso que se lleva a cabo en los diferentes centros de estudios de la provincia. La presente investigación está dirigida a desarrollar un sistema informático que permita el control de la distribución de uniforme escolar en la provincia de Granma. En ella se propone un nuevo algoritmo basado en *Q-Learning* para optimizar la planificación del proceso de elaboración de los uniformes escolares en los talleres de confecciones. El algoritmo *Q-Learning* del Aprendizaje Reforzado es una solución al problema de secuenciación de tareas con ambiente *Flow Shop* en un entorno real. El desarrollo del sistema informático se basa en tecnologías libres y multiplataforma. Las tecnologías son HTML 5, CSS 3, JavaScript, Bootstrap, jQuery y CodeIgniter. Se utilizó como metodología de desarrollo de *software Extreme*

Programming que es una metodología ágil y como patrón arquitectónico el Modelo-Vista-Controlador. Se realiza una comparación de los resultados obtenidos a partir de la ejecución del algoritmo con datos reales de la entidad. Después del análisis de las pruebas realizadas, se comprueba la utilidad y fiabilidad del software que se obtiene, contribuyendo a mejorar la distribución de uniforme escolar en la provincia de Granma.

Palabras clave: *Distribución, Uniforme escolar, Sistema informático, Planificación.*

Abstract: The distribution of school uniforms in the province of Granma is a process that is carried out in the different study centers of the province. The current research paper is aimed at developing a computer system that allows the control of the distribution of school uniforms in the province of Granma. It proposes a new algorithm based on Q-Learning to optimize the scheduling of the process of making school uniforms in clothing workshops. The Q-Learning algorithm of Reinforced Learning is a solution to the problem of sequencing tasks with a Flow Shop environment in a real context. The development of the computer system is based on free and multiplatform technologies. The technologies are HTML 5, CSS 3, JavaScript, Bootstrap, jQuery and CodeIgniter. Extreme Programming was used as an agile methodology of development of software and the Model-View-Controller as an architectural pattern. A comparison of the results obtained from the execution of the algorithm with real data of the entity is performed. After analysis of the tests carried out, usefulness and reliability of the software developed are checked, which contributes to the improvement of the distribution of school uniforms in the province of Granma.

Keywords: *Distribution, School uniform, Computer system, Scheduling.*

1. Introducción

El capital humano y las Tecnologías de Informática y las Comunicaciones (TIC) toman un nuevo significado y surge la conciencia de que son la clave para el éxito del desempeño organizacional. Uno de los aspectos sobre los que más han influido las TIC en la actualidad es la gestión de la información. Este se ha convertido en un asunto de prioridad en la mayoría de las instituciones: empresas, centros educacionales, salud, centros productivos y unidades de servicios que contribuyen al desarrollo económico de la sociedad [1].

En Cuba, unos de los sectores priorizados es el sector de la educación. En este sector se hace necesario controlar desde la gestión de información de la matrícula de los estudiantes, las calificaciones obtenidas durante el curso escolar, materiales docentes, así como la distribución de uniformes escolares. El programa para el uniforme escolar es un programa de venta regulada dirigido a los estudiantes de la enseñanza primaria, secundaria, preuniversitaria y politécnica, para garantizar la identidad en las escuelas cubanas. Esto ocurre entre el final de un curso escolar y el comienzo del otro.

La gestión de uniformes escolares en Cuba, depende de la Dirección de Comercio que cuenta con varias unidades donde se comercializa el producto. El Ministerio de Educación (MINED), que controla varias escuelas, encargadas de realizar el estudio de la cantidad de estudiantes y Confecciones Antares que tiene varios talleres donde se confeccionan los uniformes según el levantamiento realizado. En estos momentos el proceso de la distribución de uniformes escolares en la provincia de Granma se realiza de forma manual. En ocasiones se realizan informes con el procesador de textos Microsoft Word y se registran algunos datos en el tabulador electrónico Excel, pero no todos los especialistas tienen los conocimientos necesarios de informática para trabajar con estas herramientas. Todo esto dificulta la realización de una operación determinada, lo que trae consigo:

- Pérdida de información relevante por deterioro del papel con el paso del tiempo.
- Demora en los informes mensuales, debido al envío de la información desde las escuelas.
- Demora en la obtención de algún dato en específico ya que se debe consultar gran cantidad de documentos.
- Falta de seguridad de los documentos pues son vulnerables a pérdidas, alteración o sustracción.
- Demora en la planificación de la distribución de los uniformes escolares en las tiendas del municipio.

El presente trabajo tiene como objetivo general el desarrollo de un sistema informático que permita mejorar la distribución de uniforme escolar en la provincia de Granma.

Este objetivo general fue desglosado en los objetivos específicos siguientes:

- Analizar el proceso de distribución de uniforme escolar.
- Seleccionar las herramientas, lenguajes, tecnologías y metodología a utilizar para el desarrollo del software a partir del análisis efectuado.
- Diseñar el sistema informático para la distribución de uniforme escolar en la provincia de Granma.
- Implementar el sistema informático para la distribución de uniforme escolar en la provincia de Granma.
- Validar mediante pruebas de software la propuesta de solución.
- Realizar la implementación computacional del algoritmo *Q-Learning* para resolver el problema planteado.
- Proponer instancias de problemas de Secuenciamiento de Tareas o *Flow Shop Scheduling Problems* (FSSP) en un entorno real y evaluar el algoritmo con las mismas.

En su estructura el artículo está integrado por introducción, estado del arte, materiales y métodos, resultados y conclusiones. En el apartado del estado del arte se abordaron diferentes aspectos relacionados con aplicaciones que existen en la actualidad y que permiten la distribución de uniformes escolares. Se describen herramientas y lenguajes que se tuvieron en cuenta para el desarrollo de la aplicación, además se expondrán algunos ejemplos de cómo trabajar con el software y se propone un algoritmo basado en el Aprendizaje Reforzado, para optimizar el proceso de planificación de la elaboración de los uniformes escolares en los talleres de confecciones.

2. Estado del arte

En el mundo el procedimiento de entrega de paquetes escolares, se ejecuta una vez por año, iniciado con la identificación de la demanda del personal que requerirá el beneficio de uniformes. Esta detección de la demanda se efectúa a través de la información relacionada al personal administrativo, generada por parte de la Dirección de Desarrollo Humano y es validada por las diversas Direcciones del Ministerio de Educación. Es el MINED el encargado de informar la cantidad de paquetes escolares que se necesitan y seguir la Normativa general de transferencia, ejecución y liquidación de fondos para la confección de uniformes, adquisición de zapatos y paquetes de útiles escolares para estudiantes de educación parvularia y educación básica [2].

2.1. Caracterización del proceso de distribución de uniforme escolar en la provincia de Granma

En la provincia de Granma el Ministerio de Educación tiene entre sus propósitos más inmediatos actualizar el reglamento escolar. En él se acomodará lo establecido a los cambios que se desarrollan en este sector para lograr mayor calidad, y se le sumará la actualización de la Resolución 45, en la que se regula todo lo concerniente a los uniformes de los diversos niveles de enseñanza.

El proceso de distribución de uniformes escolares, es un proceso que se lleva a cabo en los diferentes centros de estudios de la provincia. En él intervienen especialistas de varios sectores que mediante la organización y el trabajo en equipo logran un eficiente trabajo. Los sectores involucrados en este proceso son:

1. El Ministerio de Educación que controla varias escuelas a través de las direcciones provinciales y municipales. Estas escuelas son las encargadas de realizar el levantamiento teniendo en cuenta el listado de los estudiantes, tipo de enseñanza, tipo de prenda asociada a esa enseñanza, así como las tallas asociadas a la prenda.
2. La Dirección de Comercio que cuenta con varias unidades donde se comercializa el producto. Este sector a través de sus direcciones provinciales y municipales recibe el levantamiento realizado por el MINED y juntos son los encargados de enviar la información necesaria a la industria para que sea importada la materia prima y se comience la elaboración del uniforme escolar. Luego de la terminación de los uniformes escolares, la Dirección de Comercio teniendo en cuenta el levantamiento realizado, distribuye los centros escolares por las diferentes unidades de comercio del municipio con la asignación de los uniformes escolares para que se proceda a la venta del mismo.

- El Grupo Empresarial de la Industria Ligera que cuenta con varias Unidades Empresariales de Base (UEB) provinciales y municipales como Confecciones Antares. Estas UEB cuentan con varios talleres que son los encargados de elaborar las prendas según el tipo y teniendo en cuenta las diferentes tallas según el levantamiento realizado.

La Figura 1 ilustra los principales sectores y ministerios que participan en el levantamiento, elaboración y distribución de uniforme escolar.



Figura 1. Programa de uniforme escolar en Cuba.

Todo este proceso, iniciado en el mes de octubre, concluye en marzo o abril con la entrega de los uniformes escolares para la distribución y venta por la Dirección de Comercio; procedimiento que está recogido en una política de gobierno conjunta entre los ministerios de Educación, de Comercio Interior y de Industria.

La Figura 2 muestra una secuencia de actividades que se llevan a cabo durante todo el proceso de distribución de uniforme escolar:



Figura 2. Proceso de distribución de uniforme actual.

2.2. Sistemas informáticos similares para la distribución de uniforme escolar

Durante el desarrollo de la investigación se realizó una búsqueda a nivel internacional y nacional sobre sistemas informáticos existentes para la distribución de uniforme escolar. A continuación, se presentan los principales resultados.

Sistema informático de registro académico y entrega de paquetes escolares. Este sistema fue desarrollado para el centro escolar católico Monseñor Esteban Alliet en el Salvador en 2011. El mismo permite llevar a cabo los módulos para el registro de matrícula, evaluación del aprendizaje de los alumnos, registro de conducta, control de paquetes escolares y realización de inventarios de alimentos. El módulo de control de paquetes escolares registra la información relacionada con las tallas de uniformes y zapatos de cada uno de los alumnos matriculados en la institución. A su vez necesita los datos de los padres de familia, la cantidad de paquetes escolares recibidos para cada uno de los ciclos, la cantidad de productos recibidos en cuanto a los paquetes alimenticios entregados por el Ministerio de Educación y los datos de los diferentes proveedores con que la institución tiene relaciones en cuanto a la adquisición de paquetes escolares [3].

Sistema informático con interfaz web para el registro académico, recurso humano, control bibliotecario y bono escolar. Este sistema fue desarrollado para el centro escolar Canton el Espino Debajo

de Zacatecoluca, departamento de la Paz en el Salvador. El sistema permite llevar a cabo los módulos para el registro académico, recurso humano, control bibliotecario y bono escolar. Dentro del módulo de bono escolar se registra la información relacionada con las tallas de uniformes y zapatos de cada uno de los alumnos matriculados en la institución [4].

Luego de un análisis de los sistemas informáticos similares estudiados, a nivel internacional, se decidió realizar una aplicación web. Además, se demuestra que estos sistemas no se pueden utilizar por las siguientes razones:

- Estos sistemas no llevan a cabo el control de la distribución de uniforme escolar.
- Estos sistemas no cumplen con el diseño responsivo, característica indispensable en la actualidad teniendo en cuenta la variedad de dispositivos que existen para visualizar las distintas aplicaciones.
- No se ajustan a los requerimientos del cliente.
- Son sistemas privativos o adaptados a las particularidades de una institución.

2.3. Lenguajes y herramientas utilizadas

HTML5, es un lenguaje diseñado para organizar y facilitar el diseño del contenido web mediante la creación de una UI (Interfaz de Usuario) estandarizada e intuitiva para lenguaje de marcación. Facilita el desarrollo de aplicaciones para diferentes navegadores, así como para dispositivos portátiles, siendo una de las tecnologías que está impulsando los avances de los servicios de computación móvil en la nube, gracias a que permite mayor flexibilidad, así como el desarrollo de sitios web emocionantes e interactivos [5]. En la presente investigación se utilizó para el diseño de formularios y validaciones de los mismos.

CSS3, es un lenguaje que se emplea para dar formato a un sitio web desde las medidas para los márgenes, hasta las especificaciones para las imágenes, textos, los nuevos selectores y propiedades [6]. Se utilizó CSS3 porque ofrece una gran variedad de opciones muy importantes para las necesidades del diseño web actual, como el diseño responsivo una de las características utilizadas en esta investigación.

JavaScript, lenguaje de programación que surgió con el objetivo inicial de programar ciertos comportamientos sobre las páginas web, respondiendo a la interacción del usuario y la realización de automatismos sencillos. Las necesidades de las aplicaciones webs modernas y el HTML5 han provocado que el uso de JavaScript haya llegado a unos niveles de complejidad y prestaciones tan grandes como otros lenguajes de primer nivel [7]. Se utilizó en la interacción con el DOM (*Document Object Model*) y en la optimización de la interacción de los usuarios con la aplicación.

PHP 7.1, es un lenguaje de programación de alto nivel diseñado originalmente para la creación de páginas web dinámicas, aquellas cuyos contenidos no es siempre el mismo. Este lenguaje es de código abierto, adecuado para desarrollo web y puede ser incrustado en código HTML. Además, es el encargado de procesar la información de formularios o enviar y recibir cookies [8].

Como *framework* se utilizaron Bootstrap 4.0 que tiene como objetivo facilitar el diseño web. Permite crear de forma sencilla webs de diseño adaptable, es decir, que se ajusten a cualquier dispositivo y tamaño de pantalla y que se visualicen de forma correcta [9]. jQuery 3.3.1, para el lenguaje JavaScript, que permite simplificar la programación en este lenguaje. Implementa una serie de clases que permiten programar sin preocuparse del navegador con el que el usuario está visualizando la página, ya que funcionan igual en todas las plataformas más habituales [10]. Se decide escoger CodeIgniter 3.1.10 para desarrollar el sistema ya que tiene una forma específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. Implementa el patrón de arquitectura Modelo-Vista-Controlador.

3. Materiales y métodos

En este epígrafe se expondrán los principales elementos que se tuvieron en cuenta para el diseño de la aplicación y se describirán algunas de las funcionalidades para la distribución de uniforme escolar utilizando la aplicación que se propone.

Este sistema fue diseñado, para facilitar la distribución de uniforme escolar de forma dinámica y agradable al usuario. Para lograrlo se emplearon algunos principios y estándares de diseño visual en las páginas web que lo

conforman. La aplicación presenta un diseño simple y sencillo. Los colores que se utilizaron fueron el blanco y azul principalmente.

Para la construcción del sistema se tomaron en cuenta algunas reglas para el diseño: un *header*, donde se muestra la información general de sistema, el nombre del sistema e imágenes y textos que muestren de manera general el contenido de la aplicación, un menú en la parte izquierda donde se encuentran los diferentes vínculos de gestión del sistema, las migas de pan, la sección del contenido donde se muestra la información que se desea buscar y un pie de página. Kent Beck expresó que el diseño adecuado para el software es aquel que: supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos [11]. A continuación, se muestra como se manifiestan los elementos del diseño, ver Figura 3.

1. Banner.
2. Menú principal.
3. Migas de pan.
4. Área de visualización del contenido.
5. Pie de página.

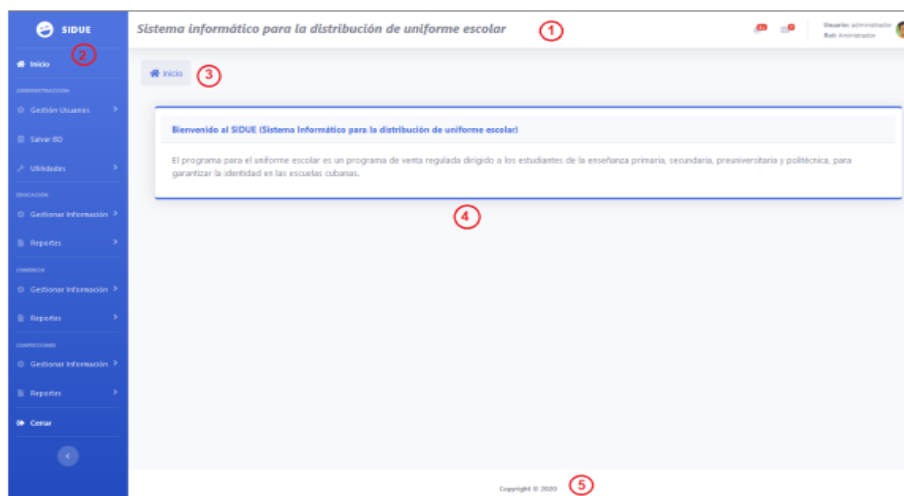


Figura 3. Ejemplo de la página Inicio del sistema. Fuente: Elaboración propia.

3.1. Arquitectura

Existen muchas definiciones de la arquitectura de software. En este trabajo se asume la siguiente definición: “la arquitectura de software es una descripción de los subsistemas y componentes de un sistema de software y las relaciones entre ellos” [12].

Dentro del estilo arquitectónico llamada y retorno se encuentra el patrón **Modelo-Vista-Controlador** (MVC): es un patrón de arquitectura de software que separa los datos y la lógica del negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello el MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. Para un mejor entendimiento de esta arquitectura se explican a continuación los tres componentes:

- Modelos: los modelos son utilizados para mantener los datos y sus reglas de negocio relevantes. Un modelo representa un solo objeto de datos. Los datos se tendrán habitualmente en una base de datos, por lo que en los modelos se encuentran todas las funciones que accederán a las tablas y harán los correspondientes *select*, *update*, *insert* y *delete* (Figura 4).

Nombre	Fecha de modifica...	Tipo	Tamaño
Colores.php	10/03/2020 22:39	Archivo de origen ...	1 KB
Comercio_ms.php	27/03/2020 10:52	Archivo de origen ...	6 KB
Comercio_ps.php	21/03/2020 22:57	Archivo de origen ...	3 KB
Educacion_ms.php	27/03/2020 14:53	Archivo de origen ...	7 KB
Educacion_ps.php	21/03/2020 20:25	Archivo de origen ...	3 KB
Ensenanzas.php	10/03/2020 22:19	Archivo de origen ...	1 KB
Escuelas.php	01/04/2020 23:56	Archivo de origen ...	10 KB
index.html	16/01/2019 16:49	Firefox HTML Doc...	1 KB
lon_auth_model.php	13/11/2019 18:25	Archivo de origen ...	65 KB
Municipios.php	11/03/2020 18:41	Archivo de origen ...	1 KB
Pedidos.php	02/05/2020 12:35	Archivo de origen ...	12 KB
Prendas.php	11/03/2020 10:55	Archivo de origen ...	2 KB
Provincias.php	11/03/2020 11:33	Archivo de origen ...	1 KB
Talles.php	10/03/2020 11:38	Archivo de origen ...	1 KB
Talleres.php	31/03/2020 23:09	Archivo de origen ...	6 KB
Tiendas.php	28/03/2020 13:24	Archivo de origen ...	7 KB
Tipos.php	11/03/2020 8:59	Archivo de origen ...	1 KB
UEBS.php	21/03/2020 18:52	Archivo de origen ...	3 KB

Figura 4. Ejemplo de la ubicación de los modelos. Fuente: Elaboración propia.

- Vistas: las vistas, como su nombre lo indica, contienen el código de la aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que permitirá renderizar los estados de la aplicación en HTML. En las vistas nada más se tienen los códigos HTML y PHP que permiten mostrar la salida (Figura 5).

Nombre	Fecha de modifica...	Tipo	Tamaño
auth	14/03/2020 16:13	Carpeta de archivos	
bienvenida	12/03/2020 21:23	Carpeta de archivos	
color	10/03/2020 22:36	Carpeta de archivos	
come_mun	25/03/2020 16:11	Carpeta de archivos	
come_prov	21/03/2020 22:28	Carpeta de archivos	
educ_mun	24/03/2020 18:47	Carpeta de archivos	
educ_prov	21/03/2020 19:41	Carpeta de archivos	
enseñanza	10/03/2020 11:36	Carpeta de archivos	
errors	16/01/2019 16:49	Carpeta de archivos	
escuela	01/04/2020 16:09	Carpeta de archivos	
login	10/03/2020 9:34	Carpeta de archivos	
municipio	11/03/2020 18:14	Carpeta de archivos	
pedido	11/04/2020 17:10	Carpeta de archivos	
prendas	10/03/2020 11:36	Carpeta de archivos	
provincia	11/03/2020 11:30	Carpeta de archivos	
talla	10/03/2020 11:36	Carpeta de archivos	
taller	31/03/2020 21:41	Carpeta de archivos	
template	10/03/2020 10:12	Carpeta de archivos	
tienda	28/03/2020 10:25	Carpeta de archivos	
tipo	10/03/2020 11:36	Carpeta de archivos	
ueb	21/03/2020 9:11	Carpeta de archivos	
index.html	16/01/2019 16:49	Firefox HTML Doc...	1 KB

Figura 5. Ejemplo de la ubicación de las vistas. Fuente: Elaboración propia.

- Controladores: contienen el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar un parte y una búsqueda de información (Figura 6).

Nombre	Fecha de modifica...	Tipo	Tamaño
Auth.php	22/03/2020 13:04	Archivo de origen ...	32 KB
Base_Dato.php	14/03/2020 23:11	Archivo de origen ...	1 KB
Color.php	14/03/2020 21:58	Archivo de origen ...	5 KB
Comercio_m.php	27/03/2020 11:53	Archivo de origen ...	9 KB
Comercio_p.php	21/03/2020 22:57	Archivo de origen ...	6 KB
Educacion_m.php	27/03/2020 11:53	Archivo de origen ...	10 KB
Educacion_p.php	21/03/2020 22:10	Archivo de origen ...	6 KB
Enseñanza.php	14/03/2020 21:59	Archivo de origen ...	5 KB
Escuela.php	02/04/2020 0:13	Archivo de origen ...	12 KB
index.html	16/01/2019 16:49	Firefox HTML Doc...	1 KB
Levantamiento.php	02/05/2020 12:39	Archivo de origen ...	10 KB
Municipio.php	21/03/2020 17:58	Archivo de origen ...	5 KB
Prenda.php	14/03/2020 22:02	Archivo de origen ...	5 KB
Provincia.php	21/03/2020 17:57	Archivo de origen ...	5 KB
Talla.php	14/03/2020 22:04	Archivo de origen ...	5 KB
Taller.php	31/03/2020 22:51	Archivo de origen ...	8 KB
Tienda_old.php	26/03/2020 9:39	Archivo de origen ...	5 KB
Tienda.php	31/03/2020 18:04	Archivo de origen ...	9 KB
Tipo.php	14/03/2020 22:05	Archivo de origen ...	5 KB
UEB.php	21/03/2020 17:57	Archivo de origen ...	6 KB
Welcome.php	25/03/2020 15:55	Archivo de origen ...	2 KB

Figura 6. Ejemplo de la ubicación de los controladores. Fuente: Elaboración propia.

3.2. Diseño de la base de datos

Durante el desarrollo de la aplicación se realizó el diseño de la base de datos, mediante la utilización de la herramienta MySQL Workbench. La misma está compuesta por 23 tablas, las cuales están normalizadas, cumpliendo con las normas establecidas para el diseño de bases de datos, ver Figura 7.

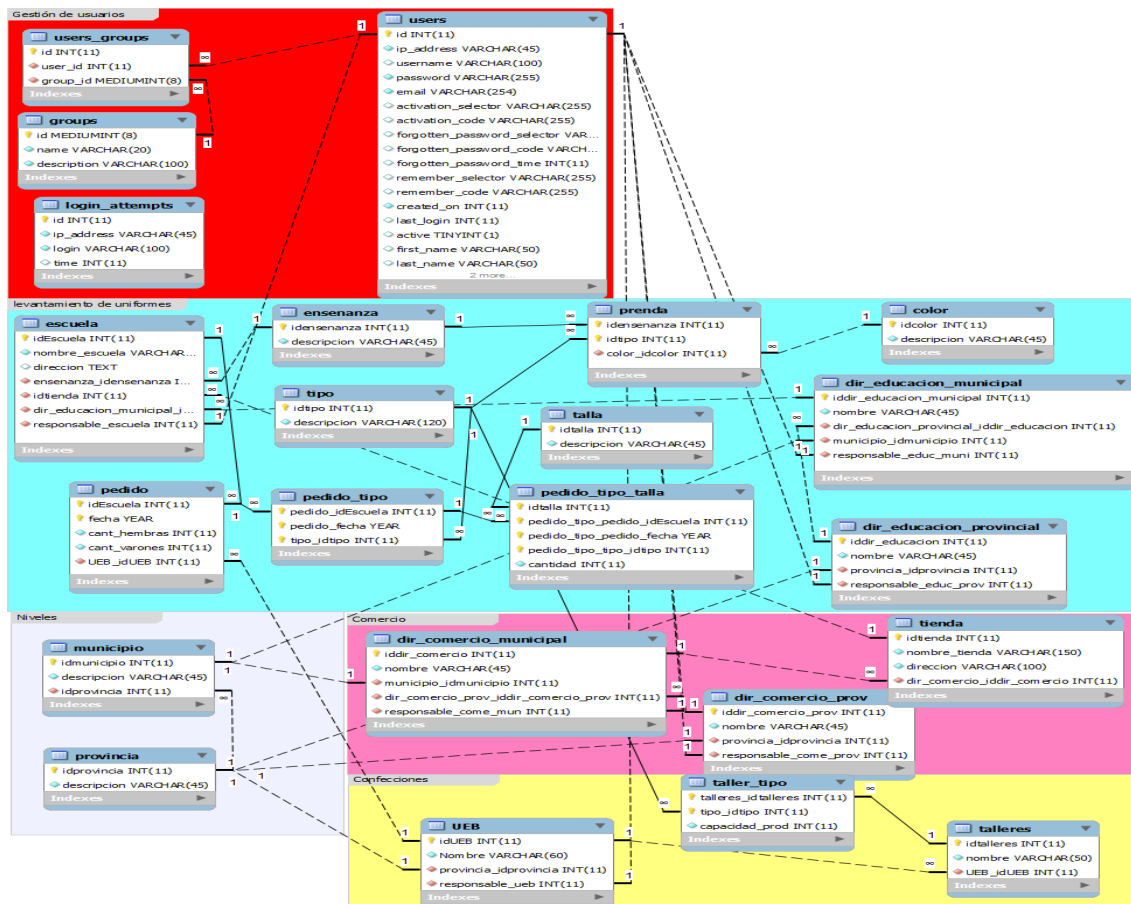


Figura 7. Modelo de la Base de datos del sistema. Fuente: Elaboración propia.

3.3. Programación de la aplicación

El software final es una aplicación web que permitirá la distribución de uniforme escolar en la provincia de Granma. Está estructurado por el área de menú donde los especialistas pueden realizar diferentes operaciones en el módulo de administración, educación, comercio y confecciones. También podrá realizar salvadas de la Base de Datos, gestionar perfil y cambiar contraseña.

Para visualizar el listado de las prendas el administrador debe acceder al menú en el módulo de administración, dar *click* en las utilidades y seleccionar la opción prenda. Se le mostrará en el área de visualización del contenido un listado de las prendas donde podrá insertar, modificar, eliminar, exportar a PDF o Excel el listado de las prendas, así como imprimir el listado, ver Figura 8.



Figura 8. Listado de prendas. Fuente: Elaboración propia.

Para realizar el levantamiento de uniforme las personas autorizadas accederán a la opción levantamiento del menú en el módulo de educación. Luego dará *click* en adicionar levantamiento de uniforme y llenará el formulario. Si el usuario introduce mal los datos, el sistema mostrará un mensaje de error y hasta que no los corrija no podrá realizar la operación. Luego debe dar *click* en el botón guardar para realizar la operación, ver Figura 9.

Figura 9. Levantamiento de uniforme escolar. Fuente: Elaboración propia.

3.4. Problema de planificación

Un problema de planificación o *scheduling* se define como el proceso de asignar recursos a actividades a lo largo del tiempo o, alternativamente, la determinación de cuándo las operaciones o acciones que componen un plan deben ser realizadas y hacer uso de dichos recursos. Adicionalmente, hay que garantizar que los tiempos asignados a dichas acciones deben cumplir con una serie de restricciones establecidas en el plan, así como con una optimización de determinados criterios. De esta forma, *scheduling* está directamente asociado con la ejecutabilidad y optimalidad de un plan preestablecido. El proceso de *scheduling* es particularmente importante en el campo de la producción y de la gestión de operaciones, y así su terminología más relevante deriva de esta fuente como por ejemplo trabajos, recursos o actividades [13,14,15].

Uno de los criterios de optimización más usual es el de la minimización del máximo tiempo de finalización del *scheduling*, conocido también como *makespan* [15].

El Aprendizaje Reforzado (*Reinforcement Learning*, RL) es un enfoque de la Inteligencia Artificial en el que los agentes aprenden a partir de su interacción con el ambiente. Es aprender qué acción tomar dada una situación determinada con el objetivo de maximizar una señal numérica de recompensa que da la medida de cuán buena fue la acción elegida por el agente [16].

3.4.1. Algoritmos del Aprendizaje Reforzado

Q-Learning es una técnica propuesta por Watkins en 1989 [17] que tiene su origen en el método *Value Iteration* y está basada en la utilización de un valor q que representa el costo esperado de seleccionar una acción en un estado determinado y a partir de ahí seguir una política óptima.

Este método es uno de los más populares del Aprendizaje Reforzado debido a que fue el primero en surgir para resolver los problemas de control, existe una prueba de la convergencia del método en [18] y en [19] y además es la extensión por excelencia del concepto de Control Óptimo, en el sentido que es una técnica simple que calcula una política óptima sin la evaluación del costo intermedio y sin usar un modelo preestablecido. Otra de las razones

de la popularidad del *Q-Learning* es que existe evidencia de que se comporta mejor que otros métodos del Aprendizaje Reforzado [20].

Este algoritmo se basa en aprender una función acción-valor que devuelva la ganancia esperada de tomar una acción determinada en cierto estado. El centro del algoritmo es una simple actualización de valores, cada par (s, a) tiene un Q -valor asociado, cuando la acción a es seleccionada mientras el agente está en el estado s , el Q -valor para ese estado-acción se actualiza basado en la recompensa recibida por el agente al tomar la acción. También se tiene en cuenta el mejor Q -valor para el próximo estado s' , la regla de actualización completa es la siguiente:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right], \quad (1)$$

En esta expresión, $\alpha \in [0,1]$ representa la velocidad de aprendizaje y r la recompensa o penalización resultante de ejecutar la acción a en el estado s . La velocidad de aprendizaje α determina el 'grado' por el cual el valor anterior es actualizado. Por ejemplo, si $\alpha = 0$, entonces no existe actualización, y si $\alpha = 1$, entonces el valor anterior es reemplazado por el nuevo estimado. Normalmente se utiliza un valor pequeño para la velocidad de aprendizaje, por ejemplo $\alpha = 0,1$. El factor de descuento (parámetro γ) toma un valor entre 0 y 1 ($0 \leq \gamma \leq 1$), si está cercano a 0 entonces el agente tiende a considerar sólo la recompensa inmediata, si está cercano a 1 el agente considerará la recompensa futura como más importante.

El algoritmo puede resumirse como sigue:

Algoritmo 2.1: Algoritmo Q-Learning.
Data: Inicializar $Q(s, a)$ arbitrariamente
Result: Valor $Q(s, a)$

```

1 Para cada episodio: Inicializar  $s$ ;
2 repeat
3   Escoger  $a$  desde  $s$  usando una política  $\epsilon$ -greedy;
4   if  $a > 0$  then
5     Observar el valor de  $r, s'$ ;
6      $Q(s, a) \leftarrow Q(s, a) + \alpha \left[ r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$ ;
7      $s \leftarrow s'$ ;
8 until condición ("s no sea terminal");
```

Figura 10. Algoritmo *Q-Learning*.

El algoritmo anterior (Figura 10) es usado por los agentes para aprender de la experiencia o el entrenamiento, donde cada episodio es equivalente a una sesión de entrenamiento.

En la presente investigación se propone el siguiente algoritmo (Figura 11), para obtener una solución óptima que contribuya a una mejor planificación de los talleres de confecciones (Figura 12). Antares en el momento de la elaboración de los uniformes escolares. El algoritmo puede resumirse como sigue:

Algoritmo 2.2: Algoritmo propuesto.
Data: Inicializar $Q(s, a) = \{\}$, $Mejor = \{\}$
Result: Orden de los trabajos

```

1 Para cada episodio:
2  $s = \{\}$ ;
3 while (existan trabajos sin procesar) do
4   Seleccionar entre los trabajos pendientes, el de mayor tiempo de procesamiento
   total  $J_m$ ;
5   Inicializar acciones como el conjunto de posibles puntos de inserción de  $J_m$  dentro
   de  $s$ ;
6   if número aleatorio  $\leq \epsilon$  then
7     Seleccionar una  $a$  al azar dentro de acciones;
8   else
9     Seleccionar la  $a$  devuelta por  $Mejor\_accion(s)$ ;
10  Tomar acción  $a$ ;
11  observar  $s'$  y  $r$  como  $\frac{1}{makespan}$ ;
12  Hacer  $a' \leftarrow Mejor\_accion(s')$ ;
13  Fin de Episodio
14   $Q(s, a) \leftarrow Q(s, a) + \alpha \left[ r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$ ;
15   $s \leftarrow s'$ ;
16  if  $makespan\ s < makespan\ Mejor$  then
17     $Mejor \leftarrow s$ ;
```

Figura 11. Algoritmo propuesto en la investigación. Fuente: Elaboración propia.

Algoritmo 2.3: Procedimiento para escoger la mejor acción.

Data: Inicializar acciones como el conjunto de posibles puntos de inserción de J_m dentro de s .

Result: Mejor acción

```

1 if  $Q$  contiene  $s$  then
2    $a = \max_a Q(s, a')$ , tal que  $a \in \text{acciones}$ ;
3 else
4   if  $a \in \text{acciones}$  then
5     Tomar acción  $a$ ;
6     observar  $s'$  y  $r$  como  $\frac{1}{\text{makespan}^t}$ ;
7   Seleccionar la  $a$  que mejor  $r$  obtenga;
8    $Q(s, a) \leftarrow r$ ;
```

Figura 12. Procedimiento para escoger la mejor acción. Fuente: Elaboración propia.

Para comprobar el correcto funcionamiento del algoritmo se seleccionó el Taller de Confecciones Antares del Nuevo Manzanillo, ubicado en el municipio Manzanillo de la provincia de Granma. En él se lleva a cabo el proceso de la elaboración de los uniformes escolares, mediante la confección de las prendas (pantalón, sayas, blusas y short) que se realizan en distintas áreas especializadas del mismo.

La cadena de producción estructurada para esta planificación de procesos requiere que cada una de las tareas a ejecutar pase por todas las máquinas en el mismo orden. Este tipo de líneas de fabricación se encuentra cada vez más en las industrias de producción reales debido a las necesidades de los clientes de tener productos específicos para ellos. Esto implica que los pedidos tienden a no ser acumulados en lotes del mismo tipo de productos y después almacenados hasta que el cliente solicita la cantidad fabricada, sino que se tiene la flexibilidad de producir diferentes tipos de productos a la vez, proveyendo al cliente con la variedad y cantidad de productos que necesita en cada momento.

Lo que se hacía en la práctica era que la persona encargada de la programación (un especialista o tecnólogo), con base a su experiencia y conocimiento del proceso productivo encontraba una solución factible al problema sin ningún tipo de ayuda computacional, utilizando lo que se conoce como las “6 Reglas de Oro” de la Planificación que en esencia no son más que secuencias fijas de las que se valen las personas encargadas de la programación dentro del taller. Una vez seleccionadas estas seis secuencias se escoge la más óptima de ellas. Ahora bien, si se parte de la base de que una computadora puede procesar miles de veces más soluciones que una persona, entonces es muy probable que con la ayuda de una técnica como los algoritmos de Aprendizaje Reforzado y Sistemas Multiagentes y el uso de un ordenador se encuentren mejores soluciones a las encontradas de forma manual por un ser humano.

La aplicación del *Q-Learning* en la solución de este problema permite que los resultados obtenidos sean superiores a los reportados en el Taller de Confecciones Antares del Nuevo Manzanillo ubicado en el municipio de Manzanillo en la provincia de Granma. Para la ejecución del algoritmo propuesto se utilizaron como datos: las máquinas (áreas por las que pasan las prendas durante la elaboración), los trabajos (tipos de prendas que se realizan en el taller relacionado con los uniformes escolares) y el tiempo que se demora realizar cada trabajo.

4. Resultados

4.1. Pruebas de software realizadas al sistema informático

Para validar el correcto funcionamiento del software se le realizaron pruebas al sistema, empleando el método de caja negra. Las pruebas realizadas al sistema fueron las pruebas de aceptación, obteniéndose los siguientes resultados:

- Se realizaron un total de 4 iteraciones de prueba para comprobar el correcto funcionamiento de las 26 Historias de Usuarios (HU).
- En la primera iteración se le realizaron las pruebas a 6 HU detectándose los siguientes errores: de ortografía en 6 HU, de usabilidad en 4 HU y de validación en 3 HU.

Tabla 1. Resultados de la primera iteración.

Errores ortográficos	HU	Errores usabilidad	HU	Errores validación	HU
En los mensajes de validación. Omisión de letras. Uso incorrecto de las tildes.	Autenticar Usuario, Gestionar Usuario, Gestionar permisos, Cambiar contraseña, Gestionar perfil y Realizar copia de seguridad de la BD.	No seguía un estándar en el uso de los botones. No se usaba las migas de pan.	Autenticar Usuario, Gestionar Usuario, Gestionar permisos y Gestionar perfil.	Permitía usuarios duplicados. No se asignaban correctamente los permisos al usuario. No tenía establecido un patrón para la contraseña.	Gestionar Usuario, Gestionar permisos y Cambiar contraseña.

Fuente: Elaboración propia.

- En la segunda iteración se le realizaron las pruebas a 12 HU, se corrigieron los errores detectados en la primera iteración y se detectaron 8 HU con errores de ortografía, 4 HU con errores de validación y 5 HU con errores de usabilidad.

Tabla 2. Resultados de la segunda iteración.

Errores ortográficos	HU	Errores usabilidad	HU	Errores validación	HU
En los mensajes de validación. Omisión de letras. Uso incorrecto de las tildes.	Gestionar tallas, prenda, enseñanza, color, tipo de prenda, provincia, municipio y UEB.	No se usaba las migas de pan.	Gestionar prenda, color, provincia, municipio y UEB.	Entrada de caracteres incorrectos.	Gestionar tallas, prenda, Dirección de Educación Municipal y UEB.

Fuente: Elaboración propia.

- En la tercera iteración se corrigieron los errores detectados en la segunda iteración y se le aplicaron pruebas a las restantes 8 HU, detectándose 2 HU con errores de usabilidad, 3 HU con errores de validación y 2 HU con errores ortográficos.

Tabla 3. Resultados de la tercera iteración.

Errores ortográficos	HU	Errores usabilidad	HU	Errores validación	HU
En los mensajes de validación. Omisión de letras. Uso incorrecto de las tildes.	Generar reporte de prendas por tienda y Gestionar talleres.	Uso incorrecto de iconos.	Realizar levantamiento de Uniformes y Gestionar tiendas.	Entrada de caracteres incorrectos y visualización de información incorrecta.	Realizar distribución de uniformes por talleres, Generar reporte de prendas por tallas y Realizar levantamiento de uniformes.

Fuente: Elaboración propia.

- En la cuarta iteración se corrigieron los errores de la iteración anterior, se probaron nuevamente las 26 HU y no se detectaron errores en las mismas.
- Las pruebas realizadas garantizaron la correcta validación del sistema informático.
- La aplicación posee una interfaz agradable y fácil para el usuario.
- La aplicación cumple con los nuevos estándares en el desarrollo web y es adaptable a cualquier tipo de dispositivo donde se visualice.

4.2. Análisis de los datos arrojados por *Q-Learning* y su presentación

A partir de los resultados arrojados por el algoritmo y los resultados reales de estas instancias dentro de la empresa se confeccionó la Tabla 4.

Tabla 4. Resultados arrojados por *Q-Learning* con los datos reales de la empresa en 20 instancias del problema.

Casos	Cantidad de Trabajos	Máquinas	Fecha	Resultados de la Empresa	<i>Q-Learning</i>
1	7	6	oct-15	206	140
2	7	6	oct-15	189	118
3	5	6	oct-15	115	95
4	6	6	oct-15	116	105
5	6	6	oct-15	110	100
6	8	6	oct-15	175	156
7	6	6	nov-15	143	110
8	5	6	nov-15	128	90
9	4	6	nov-15	105	86
10	7	6	nov-15	149	125
11	8	6	nov-15	225	163
12	7	6	dic-15	196	138
13	6	6	dic-15	122	107
14	5	6	dic-15	119	99
15	5	6	dic-15	113	89
16	6	6	ene-15	146	114
17	8	6	ene-15	246	165
18	8	6	ene-15	235	149
19	7	6	ene-15	241	142
20	6	6	ene-15	144	112

Fuente: Elaboración propia.

En la investigación se analizan tiempos de procesamiento dados en horas, con el objetivo de optimizar el tiempo de total de realizar n trabajos en m máquinas, al comparar los resultados obtenidos mediante el algoritmo propuesto con los resultados de la entidad, todos fueron menores.

La prueba U de Whitney, también llamada de Mann-Whitney-Wilcoxon, prueba de suma de rangos Wilcoxon o prueba de Wilcoxon-Mann-Whitney, es una prueba no paramétrica con la cual se identifican diferencias entre dos poblaciones basadas en el análisis de dos muestras independientes, cuyos datos han sido medidos al menos en una escala de nivel ordinal.

La prueba calcula el llamado estadístico U , cuya distribución para muestras con más de 20 observaciones se aproxima bastante bien a la distribución normal. Para el empleo de la prueba en la investigación nos planteamos las siguientes hipótesis:

- Hipótesis nula (H_0). No existen diferencias significativas entre los dos métodos.
- Hipótesis alterna (H_a). Existen diferencias significativas entre los dos métodos.

Para el procesamiento de los datos se empleó el software SPSS 15.0 para Windows. Los resultados se muestran en la Tabla 5.

Tabla 5. Estadísticos de *contraste*^b.

VAR00005	
U de Mann-Whitney	91,500
W de Wilcoxon	301,500
Z	-2,935
Sig. asintót. (bilateral)	,003
Sig. exacta [2*(Sig. unilateral)]	,003 ^a
a. No corregidos para los empates.	
b. Variable de agrupación: VAR00007	

Fuente: Elaboración propia.

Como el valor de p [Sig. asintót. (bilateral)] resultó ser menor que **0,05**, entonces se rechaza la hipótesis nula, concluyendo que existen diferencias estadísticas significativas entre los dos métodos, que permiten concluir que con los resultados obtenidos mediante el algoritmo *Q-Learning* a partir del uso del mismo, se optimiza el tiempo de elaboración de los trabajos realizados en los talleres.

4.3. Valoración y aporte

La actualidad de la investigación está dada en que se dirige hacia el perfeccionamiento del proceso de distribución escolar en la provincia de Granma, dando respuestas a limitantes existentes en la sociedad actual. Permite unificar el trabajo de los especialistas de varios sectores que intervienen en el proceso, para realizar en tiempo y forma el levantamiento y distribución de los uniformes escolares.

El aporte práctico de esta investigación se concentra en lograr relacionar de manera correcta el uso de las TIC en el proceso de distribución de uniforme escolar en la provincia de Granma, proporcionándole a los especialistas un sistema informático, donde se hace un correcto uso de algunos recursos como animaciones, imágenes y textos aprovechándolos al máximo para lograr seguridad y rapidez en los reportes que se realizan, de forma dinámica, eficiente y sencilla.

De la información como resultado del algoritmo propuesto y los datos reales del taller se pudo apreciar que a partir del uso del mismo se optimiza el tiempo de elaboración de los trabajos ya que reportó valores de *makespan* menores que los obtenidos a partir de los datos de la entidad en todas las instancias que fueron procesadas.

5. Conclusiones

La caracterización del proceso de distribución de uniforme escolar confirma la existencia de un problema científico, que precisa de la intervención de la informática para contribuir a su solución.

Se obtuvo un sistema informático que contribuye a mejorar la distribución de uniforme escolar en la provincia de Granma, a partir de la utilización de la metodología, herramientas, lenguajes, tecnologías y los estándares modernos en los diseños de aplicaciones web.

Los resultados de las pruebas de aceptación verifican la utilidad y fiabilidad del software, lo que asevera el cumplimiento del objetivo propuesto, la transformación del objeto de investigación desde la incidencia sobre el campo de acción y la contribución a la solución del problema científico; para confirmar desde estos argumentos la validez de la presente investigación.

El algoritmo propuesto permite optimizar los tiempos de elaboración de los trabajos realizados en los talleres, evidenciando que el algoritmo *Q-Learning* constituye una interesante alternativa para resolver problemas complejos de secuenciación de tareas y alcanzar resultados superiores en comparación con los existentes en la entidad.

Los autores proponen para futuras investigaciones profundizar en el proceso de distribución de uniformes escolares, con vistas a utilizar métodos, técnicas o algoritmos que permitan optimizar los tiempos de distribución en las tiendas e información en tiempo real de la disponibilidad de cada una de las prendas.

6. Referencias

- [1] Rivera, D., Suconota, E. (2018). Las TIC en la gestión de los procesos educativos. *Razón y Palabra*, 22 (3_102), 481–509. Recuperado de: <https://revistarazonypalabra.org/index.php/ryp/article/view/1278>
- [2] Cardoza, M. E. (2015). *Procedimiento para la adquisición de uniformes para el personal técnico administrativo del Ministerio de Educación*. Recuperado de: <https://www.transparencia.gob.sv>
- [3] Aguirre Monge, N. C., García Cabrera, S. G. (2011). *Sistema Informático de Registro Académico y Entrega de Paquetes Escolares para el Centro Escolar Católico Monseñor Esteban Alliet* (Tesis de pregrado). Escuela de Ingeniería de Sistemas Informáticos. Facultad de Ingeniería y Arquitectura. Universidad de El Salvador.
- [4] Avelar García, R., Guerrero, E. M., Reyes de Marquez, C. M. (2015). *Sistema Informático con Interfaz Web para el Registro Académico, Recurso Humano, Control Bibliotecario y Bono Escolar, del Centro Escolar Canton El Espino Debajo de Zacatecoluca, Departamento de La Paz* (Tesis de pregrado). Departamento de Informática. Facultad Multidisciplinaria Paracentral. Universidad de El Salvador.
- [5] LaGrone, B. (2013). *HTML5 and CSS3 Responsive Web Design Cookbook*. San Francisco, California, USA: Packt Publishing
- [6] Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y JavaScript* (3era. Ed.). México: Alfaomega.
- [7] Haverbeke, M. (2018). *ELOQUENT JAVASCRIPT. A Modern Introduction to Programming*. (3era. Ed.). Recuperado de: <http://www.eloquentjavascript.net>
- [8] Cabezas, L. (2010). *Manual imprescindible de PHP*. España: Gráficas Hermanos Gómez, S.L.L.
- [9] Bootstrap, T. (2019). *Build fast, responsive sites with Bootstrap*. Recuperado de: <http://www.getbootstrap.com/>.
- [10] Mestras, J. P. (2011). *Aplicaciones Web/Sistemas Web. Bootstrap 3.0*. Recuperado de: <https://www.lawebdelprogramador.com/pdf/14128-Bootstrap-3.0-Aplicaciones-Web-Sistemas-Web.html>
- [11] Penadés, C. (2006). *Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. Recuperado de: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
- [12] Buschmann, F., Meunier, R., Rohnert, H., Sornmerlad, P., Stal, M. (1996). *Pattern-Oriented Software Architecture: A System of Patterns*. USA: John Wiley & Sons Ltd.
- [13] Matía, J. (2007). *Optimización de la secuenciación de tareas en taller mediante algoritmos genéticos*. (Tesis de Maestría). Universidad Pontificia Comillas.
- [14] Sierra, M. R. (2009). *Mejora de algoritmo de búsqueda heurística mediante poda por dominancia. Aplicación a problemas de scheduling*. (Tesis Doctoral). Universidad de Oviedo, Oviedo.
- [15] Tarancón, S. (2009). *Supercomputing quality scheduling a soft-computing approach*. (Proyecto Fin de Máster. Universidad Complutense de Madrid, Madrid).
- [16] Sutton, R. S., Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.
- [17] Watkins, C. (1989). *Learning from delayed rewards, in Psychology Department*. (PhD Thesis). University of Cambridge.
- [18] Jaakkola, T., Jordan, M. I., Singh, S. P. (1994). On the convergence of stochastic iterative dynamic programming algorithms. *Neural Computation*, 6 (6), 1185-1201. doi: <https://doi.org/10.1162/neco.1994.6.6.1185>
- [19] Watkins, C., Dayan, P. (1992). Q-learning. *Machine Learning*, 8, 279-292. doi: <https://doi.org/10.1007/BF00992698>
- [20] Lin, L.-J. (1992). Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8, 293-321. doi: <https://doi.org/10.1007/BF00992699>