



FACULTAD 4

# **Sistema informático para el cálculo fiscal según el registro de ingresos y gastos por arrendamiento**

**Autora:** Dayanis Fundora Quintosa

**Tutores:** M.Sc. Madelis Pérez Gil

Ing. Luis Manuel Valera Pérez

**Co-tutor:** Lic. Amarilis Engracia Hernández Trujillo

La Habana, noviembre de 2023

“Año 65 de la Revolución”

## DEDICATORIA

*A mi papá que estaría muy orgulloso.*

*A mi familia.*

*Y a la nueva familia grande y maravillosa que a pesar de estar lejos su apoyo y cariño fueron indispensables.*

## **AGRADECIMIENTOS**

*Primero que todo un agradecimiento especial a los tutores por todo el apoyo y la asesoría que me han brindado, todos los consejos y tiempo que han dedicado. A todos los profesores que han estado y que me han enseñado de una forma u otra durante toda la carrera.*

*A Keydí y Alexei que han sido un gran apoyo desde el inicio.*

*Y a Liset que sin ella esto no hubiese sido posible.*

## DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmo la presente a los 7 días del mes de noviembre del año 2023.



---

Dayanis Fundora Quintosa

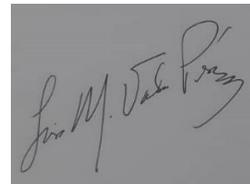
Autora



---

M.Sc. Madelis Pérez Gil

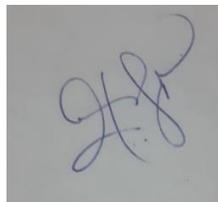
Tutora



---

Ing. Luis Manuel Valera Pérez

Tutor



---

Lic. Amarilis Engracia Hernández Trujillo

Co-tutora

## **RESUMEN**

En la actualidad los sistemas informáticos contables se han hecho más frecuentes con el objetivo de agilizar los procesos contables. Estos sistemas son de mucha utilidad para las empresas y trabajadores independientes. La presente investigación está dirigida al desarrollo de un sistema informático que, a partir del registro de los ingresos y gastos, calcule las obligaciones fiscales de los trabajadores por cuenta propia dedicados al arrendamiento de habitaciones y espacios. En la actualidad dichos registros y cálculos se realizan manualmente en tablas en excel, lo que conlleva a que el proceso sea más lento. Además, aumenta el riesgo de errores humanos que pudieran conllevar a la evasión fiscal involuntaria o aportes por encima de las cifras en perjuicio del propio arrendador. La solución a esta problemática es el desarrollo de un sistema informático que, a partir del registro de ingresos y gastos, calcule las obligaciones fiscales mensuales y anuales. Para el desarrollo de este sistema se empleó XP como metodología de desarrollo de *software*, Django 4.2.1 como marco de trabajo, PostgreSQL 13 como sistema gestor de bases de datos y Python 3.11.3 como lenguaje de programación.

Palabras clave: obligaciones fiscales, sistema informático, sistema contable, trabajador por cuenta propia

## **ABSTRACT**

*At present, computerized accounting systems have become more frequent in order to streamline accounting processes. These systems are very useful for companies and independent workers. The present research is aimed at the development of a computer system that, from the registration of income and expenses, calculates the tax obligations of self-employed workers engaged in the leasing of rooms and spaces. At present, these records and calculations are made manually in Excel tables, which makes the process slower. In addition, it increases the risk of human errors that could lead to unintentional tax evasion or contributions above the figures to the detriment of the lessor himself. The solution to this problem is the development of a computer system that, based on the recording of income and expenses, calculates the monthly and annual tax obligations. For the development of this system, XP was*

*used as the software development methodology, Django 4.2.1 as the framework, PostgreSQL 13 as the database management system and Python 3.11.3 as the programming language.*

*Keywords: accounting system, computer system, self-employed, tax obligations*

# ÍNDICE

<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	<b>5</b>
1.1 Pago de las obligaciones tributarias .....	5
1.2 Estudio del estado del arte.....	6
1.3 Metodología de desarrollo de software .....	12
1.4 Lenguajes y herramientas.....	13
Conclusiones parciales .....	17
<b>CAPÍTULO 2: ANÁLISIS Y DISEÑO DE LA PROPUESTA DE SOLUCIÓN</b> .....	<b>18</b>
2.1 Propuesta de solución.....	18
2.2 Planificación .....	19
2.3 Diseño .....	23
Conclusiones parciales .....	31
<b>CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS</b> .....	<b>32</b>
3.1 Desarrollo por iteraciones .....	32
3.2 Pruebas .....	32
3.3 Iteraciones.....	33
Conclusiones parciales .....	41
<b>CONCLUSIONES GENERALES</b> .....	<b>42</b>
<b>RECOMENDACIONES</b> .....	<b>43</b>
<b>REFERENCIAS BIBLIOGRÁFICAS</b> .....	<b>44</b>
<b>ANEXOS</b> .....	<b>46</b>
A.1. HISTORIAS DE USUARIOS.....	46
A.2. ITERACIÓN 1 .....	59
A.3. ITERACIÓN 2 .....	62
A.4. ITERACIÓN 3 .....	66
A.5. ITERACIÓN 4 .....	68
A.6. PRUEBAS DE ACEPTACIÓN - ITERACIÓN 1.....	70
A.7. PRUEBAS DE ACEPTACIÓN - ITERACIÓN 2.....	72
A.8. PRUEBAS DE ACEPTACIÓN - ITERACIÓN 3.....	74
A.9. PRUEBAS DE ACEPTACIÓN - ITERACIÓN 4.....	76

## ÍNDICE DE TABLAS

TABLA 1.1 OBLIGACIONES TRIBUTARIAS DE LOS ARRENDADORES. ....	5
TABLA 1.2 CÁLCULO DE LA BASE IMPONIBLE. ....	6
TABLA 1.3 INDICADORES A TENER EN CUENTA EN EL ESTUDIO. ....	11
TABLA 1.4 COMPARACIÓN DE LOS SISTEMAS HOMÓLOGOS. ....	11
TABLA 1.5 CARACTERÍSTICAS DE LOS SISTEMAS CONTABLES NACIONALES EXISTENTES. ....	12
TABLA 2.6 HISTORIA DE USUARIO #1. ....	20
TABLA 2.7 HISTORIA DE USUARIO #2. ....	21
TABLA 2.8 PLAN DE ITERACIONES. ....	22
TABLA 2.9 PLAN DE ENTREGA. ....	23
TABLA 2.10 TARJETA CRC #1. ....	24
TABLA 2.11 TARJETA CRC #2. ....	25
TABLA 2.12 TARJETA CRC #3. ....	25
TABLA 2.13 TARJETA CRC #4. ....	25
TABLA 2.14 TARJETA CRC #5. ....	25
TABLA 2.15 TARJETA CRC #6. ....	25
TABLA 2.16 TARJETA CRC #7. ....	26
TABLA 2.17 TARJETA CRC #8. ....	26
TABLA 2.18 TARJETA CRC #9. ....	26
TABLA 2.19 TARJETA CRC #10. ....	26
TABLA 2.20 TARJETA CRC #11. ....	26
TABLA 2.21 TARJETA CRC #12. ....	26
TABLA 2.22 TARJETA CRC #13. ....	27
TABLA 3.23 TAREA DE INGENIERÍA #1. ....	33
TABLA 3.24 TAREA DE INGENIERÍA #2. ....	33
TABLA 3.25 TAREA DE INGENIERÍA #3. ....	34
TABLA 3.26 TAREA DE INGENIERÍA #4. ....	34
TABLA 3.27 TAREA DE INGENIERÍA #5. ....	34
TABLA 3.28 PRUEBA DE ACEPTACIÓN #1. ....	35
TABLA 3.29 PRUEBA DE ACEPTACIÓN #2. ....	35
TABLA A1.30 HISTORIA DE USUARIO #3. ....	46
TABLA A1.31 HISTORIA DE USUARIO #4. ....	47
TABLA A1.32 HISTORIA DE USUARIO #5. ....	47
TABLA A1.33 HISTORIA DE USUARIO #6. ....	48
TABLA A1.34 HISTORIA DE USUARIO #7. ....	49
TABLA A1.35 HISTORIA DE USUARIO #8. ....	50
TABLA A1.36 HISTORIA DE USUARIO #9. ....	50
TABLA A1.37 HISTORIA DE USUARIO # 10. ....	51
TABLA A1.38 HISTORIA DE USUARIO # 11. ....	52
TABLA A1.39 HISTORIA DE USUARIO # 12. ....	52
TABLA A1.40 HISTORIA DE USUARIO # 13. ....	53
TABLA A1.41 HISTORIA DE USUARIO # 14. ....	53
TABLA A1.42 HISTORIA DE USUARIO # 15. ....	54
TABLA A1.43 HISTORIA DE USUARIO # 16. ....	55
TABLA A1.44 HISTORIA DE USUARIO # 17. ....	55
TABLA A1.45 HISTORIA DE USUARIO # 18. ....	55
TABLA A1.46 HISTORIA DE USUARIO # 19. ....	56
TABLA A1.47 HISTORIA DE USUARIO # 20. ....	56
TABLA A1.48 HISTORIA DE USUARIO # 21. ....	57
TABLA A2.49 TAREA DE INGENIERÍA #6. ....	59
TABLA A2.50 TAREA DE INGENIERÍA #7. ....	59
TABLA A2.51 TAREA DE INGENIERÍA #8. ....	59
TABLA A2.52 TAREA DE INGENIERÍA #9. ....	59

TABLA A2.53 TAREA DE INGENIERÍA #10.....	60
TABLA A2.54 TAREA DE INGENIERÍA #11.....	60
TABLA A2.55 TAREA DE INGENIERÍA #12.....	60
TABLA A2.56 TAREA DE INGENIERÍA #13.....	60
TABLA A2.57 TAREA DE INGENIERÍA # 14.....	60
TABLA A2.58 TAREA DE INGENIERÍA #15.....	61
TABLA A2.59 TAREA DE INGENIERÍA #16.....	61
TABLA A2.60 TAREA DE INGENIERÍA #17.....	61
TABLA A3.61 TAREA DE INGENIERÍA #18.....	62
TABLA A3.62 TAREA DE INGENIERÍA #19.....	62
TABLA A3.63 TAREA DE INGENIERÍA #20.....	62
TABLA A3.64 TAREA DE INGENIERÍA #21.....	62
TABLA A3.65 TAREA DE INGENIERÍA #22.....	63
TABLA A3.66 TAREA DE INGENIERÍA #23.....	63
TABLA A3.67 TAREA DE INGENIERÍA #24.....	63
TABLA A3.68 TAREA DE INGENIERÍA #25.....	63
TABLA A3.69 TAREA DE INGENIERÍA #26.....	63
TABLA A3.70 TAREA DE INGENIERÍA #27.....	64
TABLA A3.71 TAREA DE INGENIERÍA #28.....	64
TABLA A3.72 TAREA DE INGENIERÍA #29.....	64
TABLA A3.73 TAREA DE INGENIERÍA #30.....	64
TABLA A3.74 TAREA DE INGENIERÍA #31.....	64
TABLA A3.75 TAREA DE INGENIERÍA #32.....	65
TABLA A3.76 TAREA DE INGENIERÍA #33.....	65
TABLA A3.77 TAREA DE INGENIERÍA #34.....	65
TABLA A4.78 TAREA DE INGENIERÍA #35.....	66
TABLA A4.79 TAREA DE INGENIERÍA #36.....	66
TABLA A4.80 TAREA DE INGENIERÍA #37.....	66
TABLA A4.81 TAREA DE INGENIERÍA #38.....	67
TABLA A4.82 TAREA DE INGENIERÍA #39.....	67
TABLA A4.83 TAREA DE INGENIERÍA #40.....	67
TABLA A4.84 TAREA DE INGENIERÍA #41.....	67
TABLA A5.85 TAREA DE INGENIERÍA #42.....	68
TABLA A5.86 TAREA DE INGENIERÍA #43.....	68
TABLA A5.87 TAREA DE INGENIERÍA #44.....	68
TABLA A5.88 TAREA DE INGENIERÍA #45.....	68
TABLA A6.89 PRUEBA DE ACEPTACIÓN #3.....	70
TABLA A6.90 PRUEBA DE ACEPTACIÓN #4.....	70
TABLA A6.91 PRUEBA DE ACEPTACIÓN #5.....	70
TABLA A7.92 PRUEBA DE ACEPTACIÓN #6.....	72
TABLA A7.93 PRUEBA DE ACEPTACIÓN #7.....	72
TABLA A7.94 PRUEBA DE ACEPTACIÓN #8.....	72
TABLA A7.95 PRUEBA DE ACEPTACIÓN #9.....	73
TABLA A7.96 PRUEBA DE ACEPTACIÓN #10.....	73
TABLA A8.97 PRUEBA DE ACEPTACIÓN #11.....	74
TABLA A8.98 PRUEBA DE ACEPTACIÓN #12.....	74
TABLA A8.99 PRUEBA DE ACEPTACIÓN #13.....	74
TABLA A8.100 PRUEBA DE ACEPTACIÓN #14.....	74
TABLA A8.101 PRUEBA DE ACEPTACIÓN #15.....	75
TABLA A8.102 PRUEBA DE ACEPTACIÓN #16.....	75
TABLA A8.103 PRUEBA DE ACEPTACIÓN #17.....	75
TABLA A9.104 PRUEBA DE ACEPTACIÓN #18.....	76
TABLA A9.105 PRUEBA DE ACEPTACIÓN #19.....	76
TABLA A9.106 PRUEBA DE ACEPTACIÓN #20.....	76

TABLA A9.107 PRUEBA DE ACEPTACIÓN #21 ..... 76

## ÍNDICE DE FIGURAS

FIGURA 2.1 MODELADO DE PROCESO DEL SISTEMA.....	18
FIGURA 2.2 DIAGRAMA DE CLASES DEL SISTEMA.....	24
FIGURA 2.3 REPRESENTACIÓN ARQUITECTÓNICA DEL SISTEMA.....	28
FIGURA 3.4 RESULTADO DE LA EJECUCIÓN DE LOS TEST UNITARIOS DE LA ITERACIÓN #1.....	35
FIGURA 3.5 RESULTADOS DE LA EJECUCIÓN DE LAS PRUEBAS DE ACEPTACIÓN DE LA ITERACIÓN # 1.....	36
FIGURA 3.6 RESULTADO DE LA EJECUCIÓN DE LOS TEST UNITARIOS DE LA ITERACIÓN #2.....	37
FIGURA 3.7 RESULTADOS DE LA EJECUCIÓN DE LAS PRUEBAS DE ACEPTACIÓN DE LA ITERACIÓN #2.....	37
FIGURA 3.8 RESULTADO DE LA EJECUCIÓN DE LOS TEST UNITARIOS DE LA ITERACIÓN #3.....	38
FIGURA 3.9 RESULTADOS DE LA EJECUCIÓN DE LAS PRUEBAS DE ACEPTACIÓN DE LA ITERACIÓN #3.....	39
FIGURA 3.10 RESULTADO DE LA EJECUCIÓN DE LOS TEST UNITARIOS DE LA ITERACIÓN #3.....	40
FIGURA 3.11 RESULTADOS DE LA EJECUCIÓN DE LAS PRUEBAS DE ACEPTACIÓN DE LA ITERACIÓN #4.....	40
FIGURA A10.12 ENTREVISTA #1.....	77
FIGURA A10.13 CONTINUACIÓN ENTREVISTA #1.....	78
FIGURA A10.14 ENTREVISTA #2.....	79
FIGURA A10.15 CONTINUACIÓN ENTREVISTA #2.....	80



## INTRODUCCIÓN

En muchos países, el arrendamiento de habitaciones es una actividad cada vez más popular debido a la creciente demanda de alojamientos, especialmente en zonas turísticas. Como en cualquier actividad económica, los ingresos generados por el arrendamiento de habitaciones están sujetos a impuestos.

Las regulaciones fiscales sobre el arrendamiento de habitaciones y espacios varían según la legislación de cada país. En algunos casos, los propietarios de viviendas que alquilan habitaciones, están exentos de pagar impuestos si el ingreso generado no supera un cierto límite. En otros países se aplican impuestos sobre cualquier ingreso generado por el alquiler de habitaciones. El pago de impuestos es importante porque contribuye a la sostenibilidad del sistema fiscal, a la financiación de servicios públicos y programas sociales que benefician a toda la sociedad. El cumplimiento de las obligaciones fiscales ayuda a evitar la imposición de sanciones y multas, por parte de las autoridades fiscales (*Salazar, 2023*).

En Cuba, en 1967 se abandonó la práctica del pago de impuestos. Sin embargo, el elevado crecimiento de los gastos públicos, desde principios de los años 90 del pasado siglo, provocó la reimplantación en 1994 de un nuevo sistema tributario en correspondencia con la práctica internacional. Un eslabón fundamental a escala institucional en este proceso reformador fue la creación de la Oficina Nacional de Administración Tributaria, en lo adelante ONAT. Dicha institución cuenta con acceso a los registros y a los flujos financieros de los particulares (*López, 2012*).

En el proceso de transformación que se ha generado a partir de la actualización del modelo económico en el país, han sido diversas las reformas implementadas. Una de estas reformas ha sido el surgimiento de los trabajadores por cuenta propia, en lo adelante TCP, el cual cobra fuerza en la sociedad cubana actual.

El modelo económico cubano se orienta a la estabilidad macroeconómica, la recuperación productiva, la racionalización de las funciones del Estado y la elevación de la eficiencia en el uso de los recursos gubernamentales. Dentro del sistema tributario, el impuesto sobre las ventas constituye el principal gravamen por lo que, si se informatiza el proceso de cálculo del mismo, garantiza una mayor confiabilidad y seguridad (*Hidalgo de los Santos, 2016*).

El arrendamiento de habitaciones y espacios como actividades a ejercer por cuenta propia está regulado por el Decreto Ley 44 publicado en la Gaceta Oficial No. 94 de fecha 19/08/2021. Los propietarios de viviendas que deseen alquilar habitaciones y espacios deben registrarse ante las autoridades fiscales siguientes:

- ONAT, creada en el año 1995, mediante el Acuerdo No. 2915 del Comité Ejecutivo del Consejo de Ministros,
- Ministerio de Trabajo y Seguridad Social (MTSS), creado en el año 1974, mediante la Ley No. 1307.

En el caso de los TCP que radican en el municipio Habana Vieja, deben registrarse, en la Oficina de Facilitación al Pago de la Contribución (OFA). El monto del impuesto varía en dependencia del ingreso generado y de las regulaciones fiscales específicas de cada municipio.

Como cualquier actividad que genera ingresos los TCP deben reflejar su gestión en términos monetarios en un registro contable. La legislación vigente que instruye cómo realizar el registro contable de las operaciones de los TCP, aparece regulada en la Resolución 346/2021-“Normas específicas de contabilidad de la actividad de trabajo por cuenta propia”, emitida por el Ministerio de Finanzas y Precios.

El pago de los impuestos (tributo) se encuentra informatizado ya que se puede realizar por las plataformas de ENZONA y Transfermóvil, lo cual facilita su realización. No obstante, los procesos para obtener la cifra a pagar y los registros contables, se hacen manualmente lo que incrementa la posibilidad de incurrir en errores u omisiones.

Actualmente el registro de los datos de los clientes, de los ingresos por arrendamiento de habitaciones y espacios, así como el registro de los gastos se realiza de forma manual en un documento excel perteneciente al paquete de Office, pero esto no permite que se registren contablemente:

- los ingresos obtenidos,
- los gastos en los que se incurrieron.

Lo anteriormente expresado conlleva a que se tenga que hacer a mano en otro documento.

Para el cálculo de los impuestos y contribuciones mensuales se cumplimentan modelos en excel y mediante fórmulas que se hacen manualmente se calculan los porcentajes de los

ingresos que se deben pagar a la ONAT y a la OFA, según la legislación vigente, lo que pudiera conllevar a que:

- existan errores y omisiones que puedan provocar sanciones y multas por parte de las autoridades al afectar el resultado del cálculo de los tributos y del aporte anual al fisco,
- el arrendador no conozca diariamente los importes a pagar mensual y anualmente,
- no se puedan obtener automáticamente los estados financieros.

Los TCP dedicados al arrendamiento de habitaciones y espacios no obtienen su estado de resultado como desean ya que no tienen el conocimiento contable. En la mayoría de las ocasiones dependen de una tercera persona que los realice de forma manual y con la posibilidad real de que se los confeccionen incorrectamente, por error o intencionalmente.

Dada la situación descrita con anterioridad surge como **problema a resolver**: ¿Cómo mejorar el proceso de cálculo y registro de las obligaciones fiscales de los TCP dedicados al arrendamiento de habitaciones?

Siendo el **objeto de estudio** de esta investigación los sistemas de cálculo y registro de las obligaciones fiscales.

Para resolver el problema anterior se establece como **objetivo general**, desarrollar un sistema informático para el cálculo de las obligaciones fiscales de los TCP dedicados al arrendamiento de habitaciones y espacios.

Se define como **campo de acción** los sistemas para el registro de ingresos, cálculo de las obligaciones fiscales de los TCP dedicados al arrendamiento de habitaciones y espacios.

Para dar cumplimiento al objetivo general se derivan las siguientes **tareas de la investigación**:

1. Análisis de las herramientas y tecnologías existentes que permiten el consumo y reutilización de contenido web, para establecer similitudes con la investigación en curso.
2. Análisis de los procesos que intervienen en el consumo y reutilización de contenido web, para tener un mejor dominio del problema a resolver.
3. Estudio de los elementos correspondientes a la planificación del *software* con el objetivo de llevar a cabo un desarrollo organizado.
4. Análisis de los elementos correspondientes al diseño del *software* para guiar la implementación del mismo.
5. Estudio de las herramientas y tecnologías seleccionadas para la implementación del

sistema para lograr una aplicación funcional como solución al problema de la investigación.

6. Implementación de un sistema informático para el cálculo de las obligaciones fiscales de los TCP dedicados al arrendamiento de habitaciones.
7. Validación del correcto funcionamiento del sistema desarrollado.

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

**Métodos teóricos:**

Análítico – Sintético: se utilizó para analizar y resumir la información adquirida de bibliografías sobre sistemas para el pago de impuestos.

**Métodos Empíricos:**

Consultas bibliográficas: utilizado para la consulta de las fuentes bibliográficas durante la investigación con el objetivo de elaborar el marco teórico de la investigación.

Entrevistas: información con diversos trabajadores por cuenta propia de cómo se realiza el cálculo fiscal.

El presente informe se ha estructurado en introducción, desarrollo (tres capítulos), conclusiones y recomendaciones. A continuación, un resumen de las diferentes temáticas que se abordan en los capítulos:

**Capítulo 1: Fundamentación teórica:**

En este apartado se define el marco teórico conceptual y se realiza un estudio del estado del arte de los sistemas para el cálculo de las obligaciones fiscales. También se describen la metodología de desarrollo de *software* y las herramientas a utilizar para el desarrollo del sistema.

**Capítulo 2: Análisis y diseño de la propuesta de solución:**

En este conjunto de epígrafes se brinda una propuesta de solución basándose en el análisis del proceso de cálculo de las obligaciones fiscales de los trabajadores por cuenta propia. Se realiza una descripción detallada de dicho proceso y se modela el mismo. Se plantea la estructura modular del sistema.

**Capítulo 3: Implementación y pruebas:**

En este capítulo se describen las etapas de implementación y prueba del sistema. Se detallan elementos como las pruebas unitarias, de aceptación y las tareas de ingeniería.



	El exceso de 9 510.00 hasta 15 000.00 ----- 5% El exceso de 15 000.00 hasta 20 000.00 ----- 7.5% El exceso de 20 000.00 hasta 25 000.00 ----- 10% El exceso de 25 000.00 hasta 30 000.00 ----- 15% El exceso de 30 000.00 ----- 20%
Impuesto por la utilización de la fuerza de trabajo	Paga trimestralmente el 5% del total de las remuneraciones pagadas a sus trabajadores contratados. El importe mínimo para el cálculo mensual es el salario medio de la provincia
<b>Tasas</b>	
Tasa por radicación de anuncios	El importe a pagar lo define la ONAT a partir del tamaño y las características del negocio
<b>Contribuciones</b>	
Contribución especial a la seguridad social	Paga trimestralmente el 20% de la base de contribución seleccionada en su afiliación
Contribución a la Restauración	Paga el 10% del 5% de los ingresos totales obtenidos en el mes en caso de ser un arrendador que radica en el municipio Habana Vieja

Además, anualmente el contribuyente tiene la obligación de declarar mediante el modelo “DJ-08 Declaración Jurada” la información exigida por ley, en función de determinar la cuantía a pagar por concepto del tributo, quedando obligado con el contenido y exactitud de los datos consignados en ella y pudiendo ser sancionado conforme a derecho si no la presenta o si la presentase con inexactitud, incompleta o fraudulenta (*Ley 113 Del Sistema Tributario, de 23 de Julio Del 2012.*, n.d.).

En dicha declaración se cumplimentan de la sección A a la I haciendo cálculos de suma y resta a partir de los datos declarados mensualmente durante el año fiscal (ingresos, gastos y tributos). Además, se calcula la base imponible que es la cantidad que se usa para calcular los impuestos a pagar la misma se establece de la siguiente manera:

Tabla 1.2 Cálculo de la base imponible.

Ingreso neto anual	Tipo impositivo
El exceso de 0 hasta 10 000.00	15%
El exceso de 10 000.00 hasta 20 000.00	20%
El exceso de 20 000.00 hasta 30 000.00	30%
El exceso de 30 000.00 hasta 50 000.00	40%
El exceso de 50 000.00	50%

## 1.2 Estudio del estado del arte

En la actualidad existen una variedad de sistemas contables financieros. A continuación, se analizan los mismos, haciendo énfasis en determinados aspectos que puedan ser de utilidad para esta investigación.

### 1.2.1 Internacional

En el ámbito internacional existen una serie de sistemas informáticos relacionados con el pago de tributos, estos pueden variar significativamente de un país a otro y algunos países pueden tener sistemas tributarios que no están completamente automatizados. Además, los sistemas tributarios y los programas informáticos que se utilizan para calcular impuestos están sujetos a cambios y actualizaciones continuas en función de las leyes fiscales y las regulaciones gubernamentales. A continuación, se relacionan algunos de estos sistemas:

Agencia Tributaria en España: El *software* utilizado por la Agencia Tributaria en España se conoce como SII (Suministro Inmediato de Información). Se trata de un sistema de gestión tributaria electrónica que permite a las empresas presentar información detallada y actualizada sobre sus transacciones comerciales en tiempo real. El SII se aplica a ciertos tipos de impuestos indirectos, como el Impuesto sobre el Valor Añadido (IVA) y el Impuesto sobre Determinados Servicios Digitales (IDSD), y se utiliza para mejorar la eficiencia y la transparencia en la gestión de los impuestos (*Agencia Tributaria: Suministro Inmediato de Información (SII)*, 2023).

Plataforma e-filing en India: En la India por ejemplo la plataforma e-filing se basa en el uso de la tecnología de firma digital, que permite a los contribuyentes firmar documentos electrónicos de manera segura y autenticada. Los contribuyentes pueden presentar sus declaraciones de impuestos en línea a través de la plataforma e-filing, lo que les permite ahorrar tiempo y evitar errores en la presentación de las declaraciones. Además de la presentación de declaraciones de impuestos, el sistema e-filing también permite a los contribuyentes realizar otras actividades fiscales en línea, como la verificación del estado de sus declaraciones, la descarga de formularios fiscales y la realización de pagos de impuestos en línea.

El sistema e-filing también incluye una herramienta de verificación de impuestos, que permite a los contribuyentes verificar la autenticidad de los recibos de impuestos y otros documentos fiscales. Los contribuyentes pueden utilizar esta herramienta para evitar fraudes fiscales y verificar la exactitud de la información proporcionada por los proveedores de servicios fiscales (*Home - Central Board of Direct Taxes, Government of India*, n.d.).

Declaración en línea Corea del Sur: El sistema informático de declaración en línea en Corea del Sur es un sistema que permite a los contribuyentes presentar su declaración de impuestos y realizar otras actividades fiscales en línea. El sistema se basa en una plataforma electrónica

que permite a los contribuyentes presentar sus declaraciones de impuestos en línea de manera segura y eficiente. Los contribuyentes pueden ingresar a la plataforma a través del sitio web del servicio nacional de impuestos de corea del sur y presentar su declaración de impuestos en línea.

Además de la presentación de declaraciones de impuestos, el sistema de declaración en línea también permite a los contribuyentes realizar otras actividades fiscales en línea, como la presentación de solicitudes de devolución de impuestos y la realización de pagos de impuestos en línea (*Ganancia*, n.d.).

Odoo: Es un sistema integrado de *software* empresarial de código abierto que ofrece una amplia gama de aplicaciones y módulos para la gestión de diversas áreas de una empresa. Es conocido como un sistema de planificación de recursos empresariales (ERP) y se utiliza para administrar y coordinar las operaciones comerciales en áreas como ventas, compras, inventario, contabilidad, recursos humanos, marketing, producción y más. Una de las características distintivas de Odoo es su enfoque modular, lo que significa que las empresas pueden elegir e implementar solo los módulos que se adaptan a sus necesidades específicas. Esto permite una mayor flexibilidad y escalabilidad, ya que las empresas pueden agregar nuevos módulos a medida que crecen o adaptar el sistema a medida que cambian sus requisitos comerciales (*Desarrollador — Documentación de Odoo 16.0*, 2023). Odoo está disponible en dos versiones : Odoo Enterprise (fuentes con licencia) y Odoo Community (código abierto). En el caso del módulo Contabilidad la versión de código abierto no la tiene implementada, solo la tiene las fuentes con licencia (*Odoo Enterprise vs Community | Odoo Editions Comparison*, n.d.).

### **1.2.2 Nacional**

Con relación a los sistemas informáticos existentes en el país fueron analizados los más utilizados, los cuales se relacionan a continuación:

ASSETS NS: Es un Sistema de gestión integral, estándar y parametrizado que permite el control de los procesos de:

- Compras,
- Ventas,
- Producción,

- Taller,
- Inventario,
- Finanzas,
- Contabilidad,
- Presupuesto,
- Activos fijos,
- Útiles y herramientas,
- Recursos humanos.

Como sistema integral todos sus módulos trabajan en estrecha relación, generando, automáticamente, al subsistema de contabilidad los comprobantes de operaciones por cada una de las transacciones efectuadas, esto permite que se pueda trabajar bajo el principio de contabilidad al día, si se elige generar automáticamente los asientos diarios de cada una de las transacciones efectuadas a través del sistema en los diferentes módulos. Se podrá, además, confeccionar otros asientos diarios de operaciones no controladas por ASSETS y emitir en cualquier momento los estados financieros de la entidad en múltiples monedas atendiendo a la tasa de cambio. Los estados financieros se pueden emitir para toda la compañía, por centros de costo o agrupaciones (*Ariel Hernández Cejas, 2008.*).

VERSAT-Sarasola: El programa VERSAT-Sarasola, sistema cubano de contabilidad, permite enviar información, de forma inmediata y eficaz, desde lugares apartados, a la vez que ofrece mayor organización, control y disciplina en cada gestión. Fue éste el primer sistema de contabilidad cubano certificado, en cuya evaluación participaron el Ministerio de Finanzas y Precios, consultorías internacionales y el organismo encargado de la seguridad informática.

Es un sistema económico integrado, constituido por los módulos de:

- Configuración y seguridad,
- Contabilidad general y de gastos,
- Costos y procesos,
- Análisis económico empresarial,
- Control de activos fijos,
- Finanza y caja,

- Planificación y presupuestos,
- Control de Inventarios,
- Pago de salario,
- Paquete de gestión,
- Contratación,
- Facturación.

Elabora los estados financieros, así como estados proformas en el proceso de planificación. Analiza el resultado económico-financiero de la entidad y cada uno de los eslabones de dirección, en forma totalmente automatizada. Emite reportes por los resultados fundamentales obtenidos en cada subsistema a partir de las opciones que el usuario puede seleccionar. Actualmente lo utilizan alrededor de 200 entidades de varias provincias y en lo adelante lo introducirán más de dos mil 500 unidades presupuestadas del país, entre las que figuran organismos de la administración central del estado, las direcciones municipales de finanzas, tesorerías, la ONAT y otros (*Bermúdez, 2018*).

RODAS XXI: Es un sistema multiempresa que cuenta actualmente con seis módulos:

- Finanzas,
- Contabilidad,
- Activos Fijos,
- Nóminas,
- Inventario,
- Facturación.

Estos módulos pueden emplearse integrados en su totalidad, formando cualquier subconjunto entre ellos, o cada uno de forma independiente. Cada uno de los módulos del sistema está diseñado para trabajar en entorno de red e intercambiar información mediante correo electrónico o disquetes. Maneja un número ilimitado de empresas. Sus módulos pueden ajustarse fácilmente a las características de cada usuario. Permite el intercambio automático de los comprobantes generados por cada módulo con el de Contabilidad. Trabaja con doble moneda. La obtención de informes en este sistema se facilita por la posibilidad que brinda el mismo de establecer cinco niveles de análisis contables, contando a las cuentas principales, subcuentas y tres análisis dentro de cada subcuenta. El sistema incluye una gran variedad de

informes que le permitirán obtener desde las facturas por edades de un cliente en particular hasta el comprobante específico donde se contabilizó una factura de un proveedor, pudiendo además configurar mediante opción informes de acuerdo a sus necesidades. El sistema incluso le brinda la posibilidad de configurar a gusto del usuario informes generados en excel dentro del módulo, actualizables de forma automática y en cualquier momento, mediante los cuales pueden obtenerse desde estados financieros hasta informes de indicadores económico-financieros, y cualquier otro informe que pueda obtenerse a partir de la información contable contenida en este módulo (Citmatel, 2020, p. 0).

Se investigó acerca de 6 sistemas homólogos de ellos 3 internacionales y 3 nacionales. Fueron analizadas tres variables fundamentales e identificados aspectos negativos y positivos de estos sistemas realizando una comparación, la cual se muestra a continuación:

Tabla 1.3 Indicadores a tener en cuenta en el estudio.

Variables	Agencia Tributaria	E-filing India	Oddo	Assets NS	Versat-Sarasola
Tipo de licencia	Pago	Pago	Pago	Pago	Pago
Tipo de cálculo fiscal que realizan	No	No	No	No	No
Marco regulatorio	No	No	No	No	No

En el caso de los sistemas contables nacionales, en la tabla se refleja que no cumplen el marco regulatorio refiriéndose específicamente al del pago de las obligaciones fiscales. Debido a que los sistemas contables financieros existentes no calculan la cifra de las obligaciones fiscales a pagar por los TCP. Como tipo de licencia son sistemas de pago.

Tabla 1.4 Comparación de los sistemas homólogos.

	Sistemas internacionales	Sistemas nacionales
<b>Aspectos positivos</b>	Son sistemas web que permiten tomar elementos de diseño para el desarrollo del presente sistema.	Son aplicaciones de escritorio que permiten ver aspectos sobre la organización de algunas funcionalidades.
<b>Aspectos negativos</b>	Son sistemas que hay que comprar a precios elevados para poder usarlos.	No permiten el registro de clientes.
	No se adecuan a la legislación de Cuba.	No calculan las obligaciones fiscales a pagar por los arrendadores. No son económicos para actividades pequeñas.

Con relación a los sistemas nacionales se profundizó en sus características como por ejemplo lenguaje de programación, año de creado y modelo utilizado con el fin de elegir el lenguaje y

modelo de desarrollo más conveniente para el sistema a desarrollar, comprobando lo siguiente:

Tabla 1.5 Características de los sistemas contables nacionales existentes.

Sistema	Lenguaje de programación	Año de creado	Modelo de desarrollo
Versat-Sarasola	Delphi	2001	Cliente – Servidor
ASSETS NS	Visual Basic 6.0	1997	Cliente – Servidor

### 1.2.3 Conclusiones del estado del arte

El análisis de los diferentes sistemas para facilitar el cálculo de obligaciones fiscales a nivel nacional e internacional permitió llegar a conclusiones esenciales para la investigación. Los sistemas internacionales no cuentan con las legislaciones y procedimientos de pagos de los impuestos cubanos por lo que no se hace posible usarlos en nuestro país. Además, son sistemas de pago a los cuales no siempre se puede acceder. En el caso de los *softwares* cubanos brindan diferentes módulos para la gestión de la contabilidad, pero los TCP dedicados a la actividad de arrendamiento no lo utilizan ya que carecen de funcionalidades para ellos. Además, el precio de los mismos es elevado y para actividades pequeñas como esta no son económicos. Concluyendo que, no existe ningún sistema capaz de realizar el cálculo de los impuestos para la actividad de arrendamiento de habitaciones y espacios. Con esta investigación será realizado un sistema capaz de satisfacer las necesidades de los arrendadores.

### 1.3 Metodología de desarrollo de software

Desarrollar un buen *software* depende de un sin número de actividades y etapas. Elegir la mejor metodología para un equipo, en un determinado proyecto es indispensable para el éxito del producto. El papel de las metodologías es esencial en un proyecto.

En la actualidad existen dos grandes grupos, metodologías tradicionales y metodologías ágiles. Las primeras están pensadas para el uso exhaustivo de documentación durante todo el ciclo del proyecto. Las segundas ponen vital importancia en la capacidad de respuesta a los cambios, la confianza en las habilidades del equipo y al mantener una buena relación con el cliente (*Roberth G. Figueroa, 2019*).

#### 1.3.1 Programación Extrema

La metodología *extreme programming* o XP, es la metodología ágil más conocida. Fue desarrollada en la búsqueda por guiar equipos de trabajo pequeños o medianos, entre dos y diez programadores. La principal particularidad de esta metodología son las historias de usuario, las cuales corresponden a una técnica de especificación de requisitos; se trata de formatos en los cuales el cliente describe las características y funcionalidades que el sistema debe poseer.

En esta metodología se define la fecha de cumplimiento y el alcance de una entrega funcional. El cliente define las historias de usuario y el desarrollador con base en ellas establece las características de la entrega, costos de implementación y número de iteraciones para terminarla. Para cada iteración el cliente estipula cuales son las historias de usuario que componen una entrega funcional. Se realizan entregas pequeñas que son el uso de ciclos cortos de desarrollo, llamado iteraciones, que muestra al cliente una funcionalidad del *software* terminado y se obtiene una retroalimentación de él (*Harry Vite Cevalios, 2019*).

Algo muy característico de esta metodología es la programación en parejas, indica que cada funcionalidad debe de ser desarrollada por dos programadores. Las parejas deben cambiar con cierta frecuencia, para que el conocimiento no sea solo de una persona sino de todo el equipo. Para terminar con lo relevante de esta metodología, se presenta una etapa muy importante las cuales son las pruebas de aceptación, una vez que se ha desarrollado una funcionalidad, entra a pruebas por parte del cliente, dando su aprobación (*Harry Vite Cevalios, 2019*).

Consta de tres artefactos, las historias de usuario, las tareas de ingeniería (*Task Card*) y tarjetas CRC (Clase-Responsabilidad-Colaborador), en los cuales se profundizará en los capítulos II y III. Las fases de la metodología de programación extrema o XP se basan en 4 etapas, estas son: planificación, diseño, desarrollo y prueba. Su principal objetivo es que el equipo de desarrolladores trabaje con la mentalidad conjunta para colaborar y crear un producto de alta calidad.

#### **1.4 Lenguajes y herramientas**

Las herramientas y lenguajes de programación son fundamentales en el desarrollo de *software* y aplicaciones informáticas. Para el desarrollo de *software* existen diversas tecnologías como, por ejemplo:

- Lenguajes de programación, los lenguajes de programación son las herramientas básicas para escribir *software*. Algunos de los lenguajes más populares incluyen Java, Python, C++, C#, JavaScript, Ruby, entre otros.
- *Frameworks*, son conjuntos de herramientas y librerías que permiten el desarrollo rápido y eficiente de aplicaciones. Algunos de los más utilizados son AngularJS, React, Ruby on Rails, Laravel, Django, entre otros.
- Bases de datos, las bases de datos son herramientas esenciales para almacenar y gestionar la información de las aplicaciones. Algunas de las bases de datos más populares incluyen MySQL, Oracle, MongoDB, PostgreSQL.
- Entornos de desarrollo, los entornos de desarrollo (IDE, por sus siglas en inglés) son herramientas que proporcionan a los desarrolladores un ambiente integrado para la escritura, depuración y prueba de *software*. Ejemplos de entornos de desarrollo son: Visual Studio, Eclipse, NetBeans, IntelliJ IDEA, PyCharm, CodeBlocks, Atom, Sublime Text, Visual Studio Code, Komodo IDE.

Como lenguaje de programación se seleccionó Python debido a su facilidad de uso, amplia variedad de librerías y *frameworks*, capacidad de integración, buen rendimiento y portabilidad. Además, porque ha sido utilizado en investigaciones y desarrollo de sistemas contables anteriores obteniendo buenos resultados. Fue utilizado el *framework* Django ya que es una excelente opción para el desarrollo de aplicaciones web seguras, escalables y flexibles. Su productividad, seguridad, escalabilidad, flexibilidad, comunidad y madurez lo hacen ideal para una amplia variedad de proyectos web.

A continuación, se describen las herramientas y lenguajes informáticos seleccionados para el desarrollo de la solución.

#### **1.4.1 Python 3.11.3**

Python es un lenguaje de programación interpretado y de alto nivel que se utiliza en una amplia variedad de aplicaciones, desde el análisis de datos hasta la inteligencia artificial. Fue creado en 1991 por Guido van Rossum. Entre las características de Python se incluyen: sintaxis clara y concisa, amplia variedad de bibliotecas y módulos, multiplataforma, interactivo, orientado a objetos. Python se utiliza en una amplia variedad de aplicaciones, como el desarrollo web, el

análisis de datos, la inteligencia artificial, la automatización de procesos y la programación de videojuegos, entre otros (*Welcome to Python.Org*, n.d.).

#### **1.4.2 Framework de desarrollo web Django 4.2.1**

Django es un *framework* web de alto nivel escrito en Python que se utiliza para desarrollar aplicaciones web de manera rápida y eficiente. Django sigue el patrón de diseño Modelo-Vista-Template (MVT). Proporciona una gran cantidad de características y herramientas útiles para los desarrolladores, incluyendo un sistema de administración de bases de datos fácil de usar, un sistema de rutas y URLs para manejar las solicitudes de los usuarios, un sistema de plantillas para la creación de vistas HTML dinámicas, soporte para autenticación y autorización de usuarios, y una API de ORM (*Object-Relational Mapping*) para interactuar con bases de datos (*Django*, n.d.).

#### **1.4.3 HTML5**

HTML5 es una tecnología que permite a los desarrolladores web crear páginas web más avanzadas, interactivas y accesibles, gracias a su amplia gama de nuevas funcionalidades y mejoras en la semántica y estructuración del contenido web. Además, HTML5 es compatible con una amplia variedad de dispositivos y navegadores, lo que lo convierte en una herramienta esencial para la creación de contenido web moderno y adaptativo (*HTML 5, 2021*).

#### **1.4.4 Bootstrap 5.3**

Bootstrap es un conjunto de herramientas de diseño web que facilita la creación de sitios web adaptables a diferentes tamaños de pantalla al proporcionar un conjunto predefinido de estilos y componentes de diseño. Es de código abierto, muy popular y utilizado en todo el mundo, y puede ahorrar tiempo y esfuerzo a los desarrolladores en la creación de diseños personalizados y en la garantía de la compatibilidad con diferentes dispositivos y navegadores (*contributors, n.d.*).

#### **1.4.5 Visual Studio Code**

Fue utilizado Visual Studio Code versión 1.78.0. Es un editor de código fuente gratuito y de código abierto desarrollado por Microsoft. Es compatible con múltiples lenguajes de programación y plataformas, incluyendo Windows, macOS y Linux. Visual Studio. Algunas de las características más destacadas de Visual Studio Code incluyen: integración con Git,

depuración de código, extensiones, autocompletado de código, integración con múltiples lenguajes, multiplataforma (*Visual Studio*, n.d.).

#### **1.4.6 Postgresql 13**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando (*Zea Ordóñez et al., 2017*).

#### **1.4.7 PgAdmin 4**

Herramienta de administración de bases de datos PostgreSQL de código abierto que se utiliza para gestionar y administrar bases de datos PostgreSQL. Con pgAdmin, los usuarios pueden realizar tareas como crear y eliminar bases de datos y tablas, administrar usuarios y permisos, ejecutar consultas SQL, generar informes y visualizar datos de manera gráfica. Ofrece características avanzadas como la edición de datos, el diseño visual de tablas y la gestión de copias de seguridad y restauración (*pgAdmin - PostgreSQL Tools*, n.d.).

#### **1.4.8 Visual Paradigm 16.4**

Visual Paradigm es una herramienta de modelado de *software* y diseño de sistemas que se utiliza para crear diagramas y modelos de procesos empresariales, sistemas de *software* y bases de datos. Visual Paradigm ofrece una amplia variedad de características y herramientas, incluyendo: modelado UML, modelado de bases de datos, generación de código, colaboración en equipo, integración con herramientas de desarrollo (*Ideal Modeling & Diagramming Tool for Agile Team Collaboration*, n.d.).

#### **1.4.9 BPMN 2.0**

BPMN (*Business Process Model and Notation*) Es un estándar de modelado gráfico utilizado para representar procesos de negocio en forma de diagramas. Utiliza símbolos gráficos para representar distintos elementos del proceso, como tareas, eventos, flujos de secuencia, decisiones y distintos tipos de puertas lógicas. Estos símbolos se combinan para mostrar el flujo de trabajo y las interacciones entre las actividades ([https://themeforest.net/user/dan\\_fisher](https://themeforest.net/user/dan_fisher), n.d.).

### **Conclusiones parciales**

En virtud del estudio realizado, se puede definir que los sistemas contables financieros existentes cuentan con funcionalidades contables, pero no realizan el cálculo automático de las obligaciones fiscales a partir del registro de ingresos y gastos de los TCP dedicados a la actividad de arrendamiento. Por lo que ninguno de los sistemas existentes puede ser utilizado con el objetivo propuesto en esta investigación. A partir del análisis realizado se seleccionó para el desarrollo del sistema el uso de la metodología XP, el lenguaje de programación Python 3.11.3 con el Framework Django.

## CAPÍTULO 2: ANÁLISIS Y DISEÑO DE LA PROPUESTA DE SOLUCIÓN

### Introducción

Este capítulo describe el diseño de la propuesta de solución basada en el análisis y la problemática existente. Se expondrán los artefactos que propone la metodología de desarrollo de *software* XP. Con el desarrollo del presente capítulo se definirá el patrón arquitectónico que será utilizado, los patrones de diseño (patrones GRASP y GoF).

### 2.1 Propuesta de solución

Se propone realizar una aplicación web que pueda ser utilizada por los usuarios sin importar los niveles de conocimiento. A continuación, en la figura 2.1 se detalla el funcionamiento del sistema:

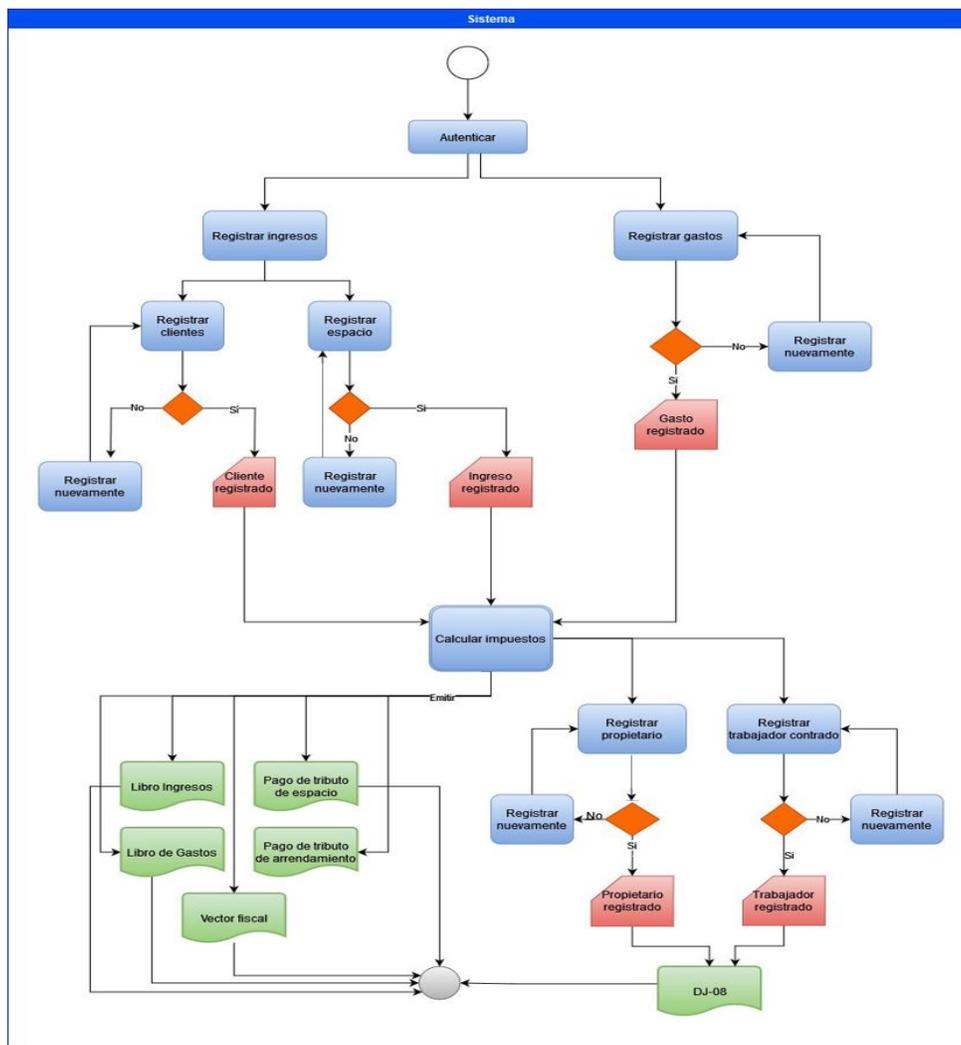


Figura 2.1 Modelado de proceso del sistema.

El sistema contará con dos roles (administrador y usuario). Para el rol administrador permitirá autenticarse y acceder a las opciones de agregar, ver, modificar y eliminar usuarios.

Para el rol usuario el sistema permitirá autenticarse y al acceder podrá registrar los ingresos y los gastos del negocio. A partir de dicho registro el sistema calculará los importes y cumplimentará los modelos establecidos por la ONAT para el pago de las obligaciones fiscales mensuales y anual. El sistema además mostrará listados resúmenes con las cifras de ingresos y gastos tanto mensuales como anuales. Calculará, además, el índice ocupacional y la disponibilidad, lo cual le permitirá al usuario y a las autoridades estatales facultadas realizar análisis sobre el funcionamiento del negocio.

## **2.2 Planificación**

La fase de planificación en la metodología XP es indispensable ya que se lleva a cabo antes de realizar el desarrollo del sistema. En esta etapa se determina el alcance del proyecto y las funcionalidades del sistema mediante las historias de usuario, el plan de iteraciones y el plan de entrega.

### **2.2.1 Historias de usuarios**

Las historias de usuarios (HU) son una técnica utilizada en esta metodología de desarrollo de *software* para describir los requisitos funcionales del sistema desde la perspectiva del usuario. Las historias de usuario se escriben en lenguaje natural. Estas historias se utilizan para priorizar el trabajo del equipo de desarrollo y para asegurarse de que el sistema se construya para satisfacer las necesidades reales del usuario (*Marcos Klender Carrasco Gonzaga, 2019*). Con el objetivo de determinar cuáles HU son más vitales resolver según las necesidades del cliente y poder realizar una correcta planificación de la implementación, cada HU es clasificada por el propietario del producto según su prioridad para el negocio en:

- Alta: HU que resultan fundamentales en el desarrollo del sistema.
- Media: HU que resultan funcionalidades necesarias, pero no imprescindibles.
- Baja: HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo, a la estructura y no tienen nada que ver con el sistema en desarrollo.

También se tiene en cuenta la dificultad y posible existencia de errores durante la implementación de cada HU, por ello el equipo de desarrollo clasifica a cada una según el riesgo en su desarrollo:

- Alto: cuando en la implementación de la HU se considera la posible existencia de errores que lleven a la inoperatividad del código.
- Medio: cuando pueden aparecer errores en la implementación de la HU que puedan retrasar la entrega de la versión.
- Bajo: cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

Las HU son representadas mediante tablas divididas por las siguientes secciones:

- Número: esta sección representa el número, incremental en el tiempo, de la historia de usuario que se describe.
- Nombre de Historia de Usuario: identifica la HU que se describe entre los desarrolladores y el cliente.
- Usuario: rol del usuario que realiza la funcionalidad.
- Prioridad en negocio: se le otorga una prioridad (Alta, Media, Baja) a las HU de acuerdo a la necesidad de desarrollo.
- Riesgo en Desarrollo: se le otorga una medida de (Alto, Medio, Bajo), a la ocurrencia de errores en el proceso de desarrollo de la HU.
- Iteración asignada: número de la iteración donde va a desarrollarse la HU.
- Puntos Estimados: es el tiempo estimado en semanas ideales (40 horas) que se demorará el desarrollo de la HU.

Fueron realizadas en total 21 historias de usuarios las cuales se detallan a continuación:

Tabla 2.6 Historia de Usuario #1.

<b>Historia de Usuario</b>	
<b>Número: 1</b>	<b>Nombre:</b> Autenticar usuario
<b>Usuario:</b> Todos los usuarios	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 0.4	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Dayanis Fundora Quintosa	
<b>Descripción:</b> Se tiene que establecer una interfaz de autenticación en la cual se solicite el nombre de usuario y la contraseña para poder acceder.	

**Observaciones:** En caso de que el dato de usuario o contraseña sea introducido incorrectamente el sistema debe notificárselo al usuario mediante un mensaje.

**Prototipo:**

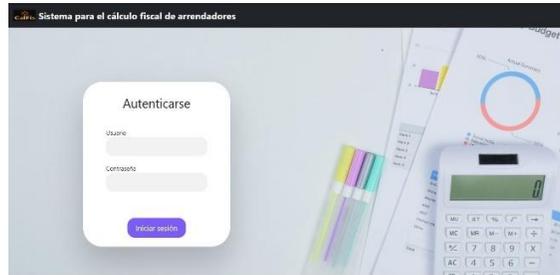


Tabla 2.7 Historia de Usuario #2.

Historia de Usuario	
<b>Número:</b> 2	<b>Nombre:</b> Gestionar usuario
<b>Usuario:</b> Administrador	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 0.6	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Dayanis Fundora Quintosa	
<b>Descripción:</b> El administrador del sistema deberá crear, eliminar, modificar y ver la información de los usuarios del sistema.	
<b>Observaciones:</b> Ninguna	
<b>Prototipo:</b>	

Véase la continuación de las HU en el Anexo 1.

### 2.2.2 Requisitos no funcionales

Los requisitos no funcionales son restricciones que especifican características del sistema que no están directamente relacionadas con sus funcionalidades específicas. Estos requisitos se centran en aspectos como el rendimiento, la usabilidad, la seguridad, la disponibilidad y otros atributos que se espera que el sistema cumpla. A continuación, se muestran los requisitos no funcionales presentes en el sistema:

#### **RNF 1. Requisitos de usabilidad:**

**RNF 1.1** La interfaz del sistema tendrá un menú principal que permita desplegar cada opción de dicho menú mostrando al darle clic los formularios y tablas seleccionadas siendo así fácil de utilizar por el usuario con una navegación clara y comprensible.

**RNF 2.** Requisitos de apariencia o interfaz externa:

**RNF 2.1** El sistema contará con una interfaz sencilla, los colores principales serán blanco, negro y gris.

**RNF 2.2** Los elementos de la interfaz como por ejemplo botones, menú, y las barras de navegación seguirán un estilo coherente en toda la aplicación.

**RNF 3.** Requisitos de software:

**RNF 3.1** El sistema deberá ejecutarse en el navegador Mozilla Firefox.

**RNF 3.2** Puede ser utilizado en sistemas operativos de Windows y Linux.

**RNF 3.3** Debe ser utilizado con el gestor de bases de datos PostgreSQL 13.

**2.2.3 Plan de iteraciones**

Con el objetivo de determinar la duración estimada del proyecto y la cantidad de puntos de historias a desarrollar, se ha establecido un plan de iteraciones. Dado que la velocidad del proyecto no se puede determinar con certeza hasta que hayan transcurrido varias iteraciones, se ha utilizado un enfoque de estimación por consenso. Este método implica que el equipo de desarrollo evalúa la complejidad y el esfuerzo requerido para completar cada HU utilizando técnicas en las que los miembros asignan puntos de historia de manera relativa, comparando la complejidad de una historia con otras previamente estimadas.

Inicialmente, se ha asignado un promedio de 4 puntos a cada iteración. Con base en esta estimación, las 21 HU se han agrupado en cuatro iteraciones. El resultado de este proceso se muestra en la tabla 2.8, donde se presenta el plan detallado de las iteraciones y las HU asignadas a cada una de ellas.

Tabla 2.8 Plan de iteraciones.

Iteración	Historias de Usuario		Duración (Puntos)
1	1	Autenticar usuario	0.4
	2	Gestionar usuario	0.6
	3	Gestionar cliente	1.1
	4	Gestionar el ingreso por arrendamiento de espacios	0.9
	5	Gestionar las bonificaciones por pago electrónico	0.6
2	6	Gestionar gastos por arrendamiento de espacios	1.1
	7	Gestionar gastos por arrendamiento de habitaciones	1.1

	8	Gestionar propietario	0.6
	9	Gestionar trabajadores contratados	0.6
	10	Calcular índice ocupacional	0.4
3	11	Calcular la disponibilidad de arrendamiento por habitaciones	0.3
	12	Calcular la disponibilidad de arrendamiento por camas	0.3
	13	Calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones	1
	14	Calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios	1
	15	Listar los ingresos totales	0.4
	16	Listar gastos de salario	0.4
	17	Listar gastos totales	0.4
4	18	Cumplimentar libros de ingresos y gastos por el arrendamiento de espacios y habitaciones	1
	19	Cumplimentar vector fiscal	0.6
	20	Cumplimentar declaración jurada	1
	21	Mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día	0.1
			<b>13.9</b>

### 2.2.4 Plan de entrega

Con el objetivo de mantener involucrado al cliente en la mejora continua del sistema se realiza un plan de entrega. A continuación, se detalla el plan de entrega del proyecto:

Tabla 2.9 Plan de entrega.

Iteración	Fecha de entrega
Iteración 1	26 de junio del 2023
Iteración 2	24 de julio del 2023
Iteración 3	14 de agosto del 2023
Iteración 4	08 de septiembre del 2023

## 2.3 Diseño

Un diseño adecuado de un *software* es el que supera las pruebas, tiene la menor cantidad de clases y métodos posibles. El diseño de *software* en XP se basa en la filosofía de desarrollo ágil y se enfoca en la simplicidad, la comunicación y la retroalimentación constante. El equipo de XP utiliza prácticas como el diseño incremental, la refactorización continua, las pruebas unitarias y el diseño emergente para desarrollar un *software* de alta calidad y fácil de mantener. Todo el diseño se centra en la iteración actual y no se hace nada anticipadamente para necesidades futuras. El resultado es un proceso de diseño disciplinado, que combina la disciplina con la adaptabilidad de una manera que la hace una de las más desarrolladas entre todas las metodologías ágiles (Roberth G. Figueroa, 2019).

### 2.3.1 Tarjetas CRC

De acuerdo a las buenas prácticas de la metodología XP, las tarjetas CRC (clase, responsabilidad y colaboración) son herramientas usadas para el desarrollo de sistemas orientadas a objetos que permiten interpretar el comportamiento de las clases, enfocándose principalmente en las tablas de las base de datos (Alexandra & Santiago, 2019).

A continuación, se muestra el diagrama de clases que representa los modelos del sistema en la figura 2.2. Este diagrama proporciona una representación visual de las clases del sistema, sus atributos y las relaciones entre ellas.

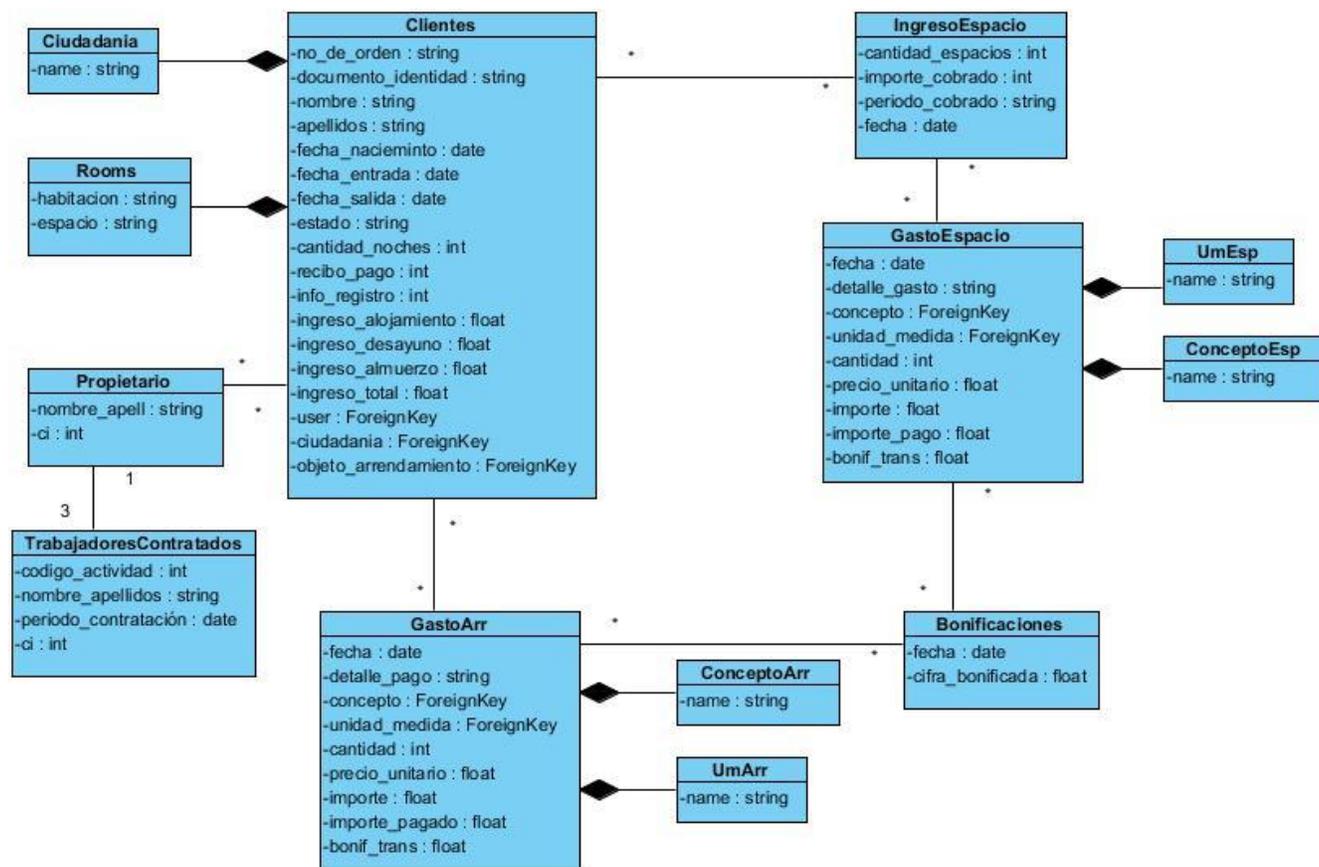


Figura 2.2 Diagrama de clases del sistema

Después de presentar el diagrama de clases, se puede proceder a describir en detalle cada una de las clases utilizando las tarjetas CRC. A continuación, se presentan las tablas con las descripciones de las tarjetas CRC correspondientes a cada clase del sistema.

Tabla 2.10 Tarjeta CRC #1.

Tarjeta CRC
<b>Clase:</b> Ciudadanía

<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a las ciudadanías del sistema.	Clientes

Tabla 2.11 Tarjeta CRC #2.

<b>Tarjeta CRC</b>	
<b>Clase:</b> Rooms	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los objetos de arrendamiento del sistema.	Clientes

Tabla 2.12 Tarjeta CRC #3.

<b>Tarjeta CRC</b>	
<b>Clase:</b> Clientes	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los clientes.	Propietario Gasto Arrendamiento Ingreso espacios

Tabla 2.13 Tarjeta CRC #4.

<b>Tarjeta CRC</b>	
<b>Clase:</b> Ingreso espacios	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los espacios.	Cliente Gasto espacio

Tabla 2.14 Tarjeta CRC #5.

<b>Tarjeta CRC</b>	
<b>Clase:</b> Gasto arrendamiento	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los gastos por arrendamiento.	Cliente Concepto arrendamiento Unidad de medida arrendamiento Bonificaciones

Tabla 2.15 Tarjeta CRC #6.

<b>Tarjeta CRC</b>	
<b>Clase:</b> Concepto arrendamiento	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los conceptos por arrendamiento.	Gasto arrendamiento

Tabla 2.16 Tarjeta CRC #7.

Tarjeta CRC	
<b>Clase:</b> Unidad de medida arrendamiento	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a las unidades de medida de arrendamiento.	Gasto arrendamiento

Tabla 2.17 Tarjeta CRC #8.

Tarjeta CRC	
<b>Clase:</b> Gasto de espacios	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los gastos por el arrendamiento de espacios.	Ingreso espacios Bonificaciones Concepto espacios Unidad de medida espacios

Tabla 2.18 Tarjeta CRC #9.

Tarjeta CRC	
<b>Clase:</b> Concepto espacios	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los conceptos por espacios.	Gasto de espacios

Tabla 2.19 Tarjeta CRC #10.

Tarjeta CRC	
<b>Clase:</b> Unidad de medida espacios	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información referente a los conceptos por espacios.	Gasto de espacios

Tabla 2.20 Tarjeta CRC #11.

Tarjeta CRC	
<b>Clase:</b> Bonificaciones	
<b>Responsabilidad</b>	<b>Colaboración</b>
Almacenar la información relacionada con las bonificaciones por pago electrónico.	Gasto de espacios Gasto arrendamiento

Tabla 2.21 Tarjeta CRC #12.

Tarjeta CRC	
<b>Clase:</b> Propietario	

Responsabilidad	Colaboración
Almacenar la información relacionada con el propietario del negocio.	Cliente

Tabla 2.22 Tarjeta CRC #13.

Tarjeta CRC	
<b>Clase:</b> Trabajadores contratados	
Responsabilidad	Colaboración
Almacenar la información relacionada con los trabajadores contratados en el negocio.	Propietario

### 2.3.2 Patrón arquitectónico

Un patrón arquitectónico es una solución generalizada y reutilizable para un problema de diseño arquitectónico común en un sistema de *software*. Los patrones arquitectónicos proporcionan un conjunto de reglas y directrices para organizar el *software* en diferentes componentes y cómo estos componentes interactúan entre sí para cumplir con los requisitos de calidad, rendimiento, seguridad, entre otros.

Debido a que el lenguaje de programación es Python y el *framework* Django el patrón arquitectónico utilizado es: Modelo-Vista-Template (MVT), el cual separa la lógica de la aplicación y la presentación de la interfaz de usuario en tres componentes diferentes: Modelo, Vista y Template. A continuación, se definen cada uno de los componentes:

- **Modelo (*Model*):** Es la capa que se encarga de manejar los datos y la lógica relacionada con ellos. En esta capa se definen las tablas de la base de datos y los métodos necesarios para interactuar con ellas.
- **Vista (*View*):** Es la capa que se encarga de manejar la lógica de negocio de la aplicación. En esta capa se definen los métodos que procesan las solicitudes del usuario y se comunican con el modelo y el template.
- **Plantilla (*Template*):** Es la capa que se encarga de manejar la presentación de los datos al usuario. En esta capa se definen las plantillas HTML que se utilizan para renderizar las respuestas que se envían al navegador del usuario (Vincent, 2020).

El patrón MVT (Modelo-Vista-Template) es una variante del patrón MVC (Modelo-Vista-Controlador) que se utiliza en Django para desarrollar aplicaciones web. En MVT, el modelo se encarga de la representación de los datos y la lógica relacionada con ellos, la vista maneja la lógica de negocio y el template es responsable de la presentación visual de los datos al

usuario. A diferencia de MVC, en MVT el template es responsable de la presentación en lugar del controlador. Este patrón MVT se utiliza para desarrollar aplicaciones web robustas y escalables con Django.

A continuación, en la figura 2.3 se muestra la representación arquitectónica del sistema:

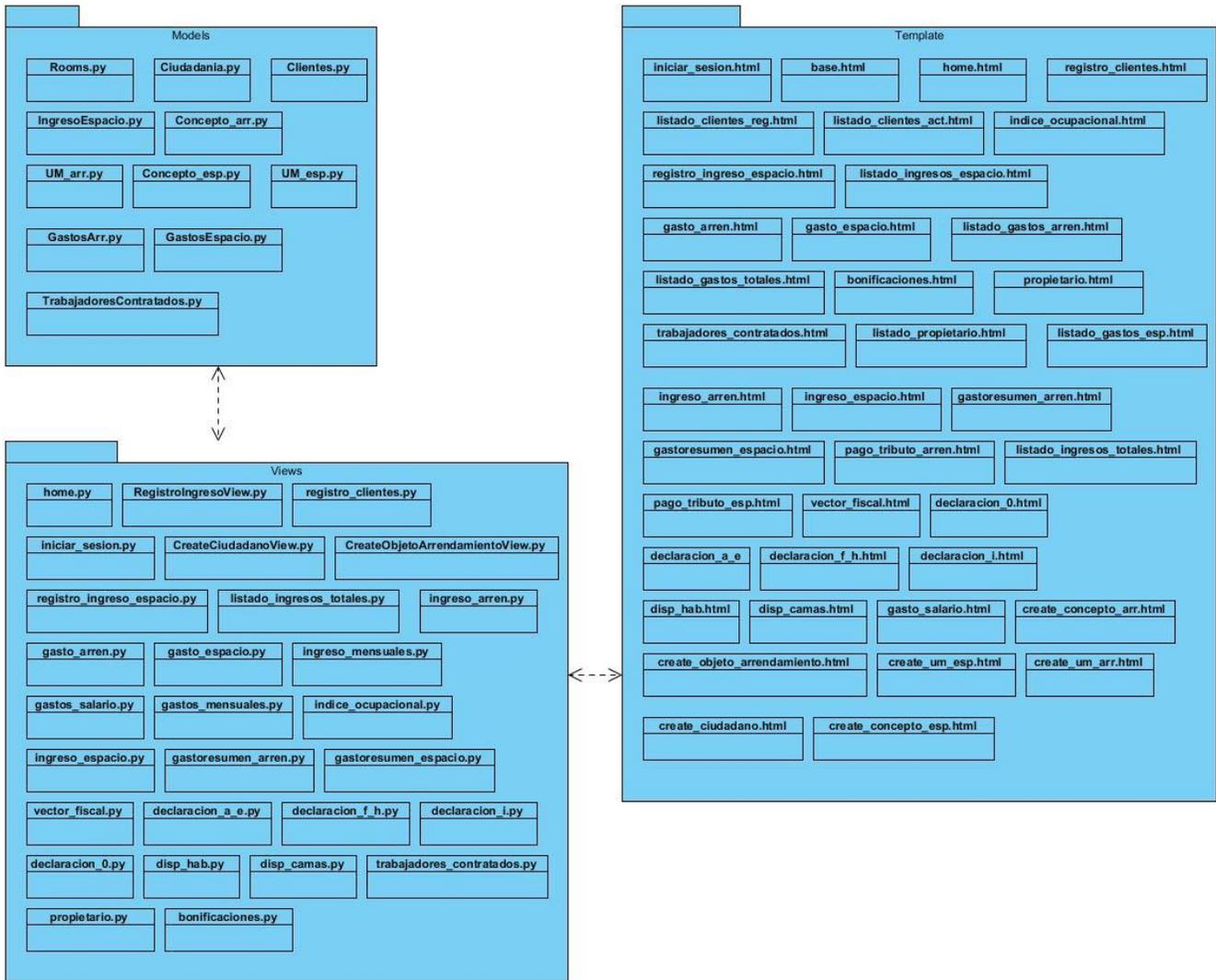


Figura 2.3 Representación arquitectónica del sistema.

### 2.3.3 Patrones de diseño

Un patrón de diseño puede ser caracterizado como una regla de tres partes, la cual expresa una relación entre cierto contexto, un problema y una solución. Para el diseño de *software*, el contexto permite a quien usa el patrón entender el entorno en el cual reside el problema y qué solución pudiera ser apropiada dentro de ese entorno. Un conjunto de requisitos, incluyendo

limitaciones y restricciones, actúan como un sistema de fuerzas que influencia en cómo el problema puede ser interpretado dentro de su contexto y cómo la solución puede ser aplicada efectivamente (*Pressman - Ingeniería.Del.Software.7ma.Ed.Docx*, n.d.).

### **Patrones GRASP**

Los patrones GRASP (*General Responsibility Assignment Software Patterns*) son un conjunto de patrones de diseño de *software* que se centran en asignar responsabilidades a los objetos en un sistema de *software*. Los patrones GRASP se basan en principios de diseño orientado a objetos y se utilizan para resolver problemas comunes en el diseño de *software* (*Pressman - Ingeniería.Del.Software.7ma.Ed.Docx*, n.d.).

A continuación, se reflejan los patrones GRASP empleados en el diseño del sistema:

**Patrón Experto:** Permite mantener el encapsulamiento de la información, puesto que los objetos utilizan su propia información para llevar a cabo las tareas asociadas a este. Por ejemplo, la clase *CreateCiudadanoView* tiene la responsabilidad de crear una vista para crear un nuevo ciudadano. Como parte de esta responsabilidad, la clase tiene acceso al modelo *Ciudadanía* y puede obtener todos los ciudadanos existentes a través de *Ciudadania.objects.all()*. Luego, agrega esta lista de ciudadanos al contexto que se pasa a la plantilla. Por lo tanto, la clase *CreateCiudadanoView* actúa como una clase experta al tener el conocimiento necesario para realizar estas tareas relacionadas con la creación de ciudadanos.

**Patrón Creador:** Las clases que tienen la responsabilidad de crear objetos contienen toda la información necesaria para construir los mismos. Por ejemplo, se crean objetos de las clases *Concepto\_arr* y *UM\_arr* utilizando el método *all()*, que devuelve todos los objetos de la base de datos de sus respectivas clases. Esto significa que la responsabilidad de crear estos objetos recae en las clases *Concepto\_arr* y *UM\_arr*.

**Patrón Alta Cohesión:** Se aplica en la mayoría las clases del diseño, ya que en cada una solo se implementan las funcionalidades que le corresponden. El patrón alta cohesión se puede evidenciar en las plantillas encargadas de mostrar los datos al usuario y en las clases manejadoras de los formularios. Por ejemplo, la clase *RegistroIngresoView* muestra la alta cohesión ya que tiene la responsabilidad de manejar la creación de registros de clientes, así como la personalización de los mensajes de éxito y error. Cada aspecto de la clase está relacionado con su propósito principal.

**Patrón Controlador:** Está representado por una clase a la cual se le asigna la responsabilidad de las operaciones del sistema, ofrece también una guía para tomar decisiones apropiadas. Este patrón se refleja en la clase *views* que es la encargada de controlar las acciones que realiza el usuario con la interfaz de la aplicación y dar respuesta a las peticiones realizadas. Por ejemplo, la función *home* actúa como el controlador principal al manejar las solicitudes del usuario y coordinar las acciones necesarias para mostrar la página de inicio y redirigir si es necesario.

**Patrón Bajo Acoplamiento:** Cada clase se comunica con un número mínimo de clases posible. El patrón bajo acoplamiento se pone de manifiesto en la relación presente entre las clases del sistema, al establecer un número mínimo de dependencias entre estas. Por ejemplo, la función *gastos\_mensuales* se utiliza como un mediador central que coordina las interacciones entre los diferentes componentes involucrados en el cálculo de los gastos mensuales. Esto ayuda a reducir la dependencia directa entre los componentes y promueve un acoplamiento bajo.

### **Patrones GoF**

Los patrones GoF (*Gang of Four*) son un conjunto de patrones de diseño de *software*. Estos patrones se utilizan para resolver problemas comunes en el diseño de *software* y se dividen en tres categorías principales: patrones creacionales, patrones estructurales y patrones de comportamiento. (*Pressman - Ingeniería. Del. Software. 7ma. Ed. Docx*, n.d.)

A continuación, se reflejan los patrones GoF empleados en el diseño del sistema:

**Patrón Controlador Frontal:** Django posee una implementación de Controlador Frontal que despacha las peticiones hacia métodos o clases, que en la práctica son páginas controladoras. Antes del despacho, la petición es procesada por varios filtros (middlewares). Por ejemplo, la lista de rutas definidas en *urlpatterns* se encarga de despachar las peticiones HTTP hacia las funciones o clases que actúan como controladores en la aplicación. Cada ruta está asociada a una vista específica que se encarga de manejar la petición y retornar una respuesta al usuario.

**Patrón Decorador:** Añade responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades. Por ejemplo, el método *registro\_clientes* se puede considerar una

forma simplificada de decoración al agregar las instancias de las clases *Rooms* y *Ciudadanía* al contexto del formulario de registro de clientes. Al agregar estas instancias al contexto, se proporciona información adicional al formulario y se enriquece la funcionalidad del objeto *ClientesForm*. Aunque no se utiliza una sintaxis específica del patrón “Decorador”, se logra un efecto similar al extender el comportamiento del formulario base con información adicional.

**Patrón Mediador:** Encargado de manejar la interacción entre los diferentes subsistemas. Realiza el mapeo objeto–relacional (ORM) a cargo del motor de Django. Simplifica y facilita los resultados de las consultas realizadas a la base de datos, así como brinda varias funcionalidades adicionales, que consultas SQL tradicionales no proveen. Por ejemplo, la función *gastos\_mensuales* actúa como un mediador central que coordina las consultas a la base de datos y los cálculos necesarios para obtener los gastos mensuales. Esto ayuda a reducir la dependencia directa entre los diferentes componentes ya que, al utilizar este patrón, se evita que los diferentes componentes de un sistema interactúen directamente entre sí. En su lugar, se utilizan intermediarios o mediadores para coordinar las interacciones entre los componentes. Esto reduce la dependencia directa entre los componentes y promueve un acoplamiento bajo, lo que facilita la modificación y extensión del código.

### **Conclusiones parciales**

En virtud de lo estudiado el desarrollo del sistema informático tendrá una duración estimada de 14 semanas. Se deben realizar 21 historias de usuario mediante las cuales quedan descritos los requisitos funcionales del sistema desde la perspectiva del usuario. Con la definición de estas historias el equipo de desarrollo puede organizar el trabajo y comenzar el desarrollo priorizando las de mayor importancia.

## **CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS**

### **Introducción**

En el presente capítulo se comenzará la fase de implementación y prueba en la cual serán ejecutadas pruebas unitarias y de aceptación. Estos procesos serán muy importantes ya que reflejan el correcto funcionamiento del sistema y la satisfacción o no del cliente con el producto entregado.

### **3.1 Desarrollo por iteraciones**

En la fase de implementación y pruebas se realizan las iteraciones planificadas. Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para cada una de estas. Al inicio de las mismas, se lleva a cabo una revisión del plan de iteraciones y se modifica de ser necesario. Como parte de este plan, se descomponen las HU en TI o tareas de ingeniería. Estas tareas son para el uso de los programadores, pueden escribirse utilizando un lenguaje técnico y no necesariamente deben ser entendibles para el cliente (*BECK, 2000*).

Antes de implementar las tareas, se elaboran los test unitarios para cada una de ellas, los cuales prueban la funcionalidad y detallan los fallos de la misma. Una vez culminada la iteración se realiza la entrega de una versión del producto al propietario del *software*, y se ejecutan las pruebas de aceptación, validando la satisfacción del cliente con el sistema. Ajustándose a la planificación realizada, se llevaron a cabo cuatro iteraciones de desarrollo sobre el sistema, obteniéndose al finalizar un producto listo para su despliegue.

### **3.2 Pruebas**

Las pruebas de *software* son una parte fundamental del proceso de desarrollo de *software*. Consisten en verificar y validar el funcionamiento de un programa para asegurarse de que cumpla con los requisitos establecidos y se comporte correctamente. En la metodología XP se utilizan principalmente dos métodos de prueba: pruebas unitarias y pruebas de aceptación.

Las pruebas unitarias verifican la funcionalidad de una sección específica del código. En la programación orientada a objetos es a una clase o a un método. Este tipo de prueba generalmente suelen ser implementadas por los desarrolladores. Las pruebas de aceptación son un tipo de prueba que se realiza para determinar si un sistema cumple con los criterios de aceptación establecidos por el cliente. Estas pruebas se centran en el comportamiento externo

del software validando que el sistema funcione de acuerdo con los requisitos del usuario y que satisfaga sus necesidades. Prueba la funcionalidad y la interacción del sistema con el usuario. El objetivo es verificar si el *software* cumple con los requisitos funcionales especificados (*Paul Ammann, 2017*).

Durante el desarrollo de cada una de las iteraciones del presente documento se realizaron pruebas unitarias y de aceptación. Las mismas fueron realizadas obteniéndose un resultado final que se muestra en las próximas secciones.

### 3.3 Iteraciones

A continuación, se detallan cada una de las iteraciones, las tareas de ingeniería y las pruebas realizadas.

#### Iteración 1

Durante esta iteración se implementaron las HU: Autenticar usuario, Gestionar usuario, Gestionar clientes, Gestionar ingresos por arrendamiento de espacios, Calcular índice ocupacional. Para su implementación las mismas fueron divididas en tareas de ingeniería (TI).

#### HU1: Autenticar usuario

Tabla 3.23 Tarea de ingeniería #1.

Tarea	
<b>Número de la tarea:</b> 1	<b>Número de Historia de usuario:</b> 1
<b>Nombre de la tarea:</b> Autenticar usuario	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos de estimación:</b> 0.4
<b>Fecha de inicio:</b> 29 de mayo de 2023	<b>Fecha de fin:</b> 31 de mayo de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla y vista para autenticar usuario, manejar mensaje en caso de autenticación fallida.	

#### HU2: Gestionar usuario

Tabla 3.24 Tarea de ingeniería #2.

Tarea	
<b>Número de la tarea:</b> 2	<b>Número de Historia de usuario:</b> 2
<b>Nombre de la tarea:</b> Listar usuarios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 1 de junio de 2023	<b>Fecha de fin:</b> 1 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Listar usuarios registrados utilizando las plantillas de Django.	

Tabla 3.25 Tarea de ingeniería #3.

Tarea	
<b>Número de la tarea:</b> 3	<b>Número de Historia de usuario:</b> 2
<b>Nombre de la tarea:</b> Adicionar usuarios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 2 de junio de 2023	<b>Fecha de fin:</b> 2 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Adicionar un nuevo usuario, mostrar los campos del formulario de registro utilizando la plantilla de Django.	

Tabla 3.26 Tarea de ingeniería #4.

Tarea	
<b>Número de la tarea:</b> 4	<b>Número de Historia de usuario:</b> 2
<b>Nombre de la tarea:</b> Editar usuario	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 3 de junio de 2023	<b>Fecha de fin:</b> 3 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Editar usuario, mostrar los campos del formulario de registro utilizando la plantilla de Django.	

Tabla 3.27 Tarea de ingeniería #5.

Tarea	
<b>Número de la tarea:</b> 5	<b>Número de Historia de usuario:</b> 2
<b>Nombre de la tarea:</b> Eliminar usuario	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 4 de junio de 2023	<b>Fecha de fin:</b> 4 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Eliminar usuario utilizando la plantilla de Django.	

Véase la continuación de las tareas de ingeniería en el Anexo 2.

## Test Unitarios

A continuación, en la figura 3.4 se muestra el resultado de los test unitarios de la iteración #1

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Destroying test database for alias 'default'...
PS C:\Users\DayiFQ\Desktop\tesis\conarren> py .\manage.py test
Found 6 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.....
-----
Ran 6 tests in 2.217s

OK
Destroying test database for alias 'default'...
PS C:\Users\DayiFQ\Desktop\tesis\conarren> 

```

Figura 3.4 Resultado de la ejecución de los test unitarios de la iteración #1.

### Pruebas de aceptación:

Tabla 3.28 Prueba de aceptación #1.

Caso de prueba de aceptación	
<b>Código:</b> HU1_P1	<b>Historia de Usuario:</b> 1
<b>Nombre:</b> Autenticar usuario	
<b>Descripción:</b> Prueba para la funcionalidad autenticar usuario.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente registrado. El usuario y la contraseña deben ser válidos.	
<b>Pasos de ejecución:</b> Se intenta autenticar un usuario en el sistema con los datos válidos.	
<b>Resultados esperados:</b> El usuario se autentica correctamente en el sistema.	

Tabla 3.29 Prueba de aceptación #2.

Caso de prueba de aceptación	
<b>Código:</b> HU2_P2	<b>Historia de Usuario:</b> 2
<b>Nombre:</b> Gestionar usuario	
<b>Descripción:</b> Prueba para la funcionalidad gestionar usuario.	
<b>Condiciones de ejecución:</b> El administrador debe estar previamente autenticado. El usuario y la contraseña deben ser válidos.	
<b>Pasos de ejecución:</b> Se intenta registrar un usuario en el sistema con los datos válidos. Se intenta registrar un usuario en el sistema con los datos inválidos. Se intenta registrar un usuario existente en el sistema. Se intenta modificar un usuario registrado en el sistema con datos válidos. Se intenta modificar un usuario registrado en el sistema con datos inválidos. Se intenta eliminar un usuario del sistema.	

**Resultados esperados:** Si los datos son correctos, se registra, se modifica y se elimina en cada caso. Si los datos son incorrectos o el usuario ya existe se redirecciona para la misma página mostrando los errores. En cada caso el sistema muestra mensajes que reflejan lo ocurrido.

Véase la continuación de las pruebas de aceptación en el Anexo 6.

A continuación, en la figura 3.5 se muestran los resultados de la ejecución de las pruebas de aceptación de la iteración #1

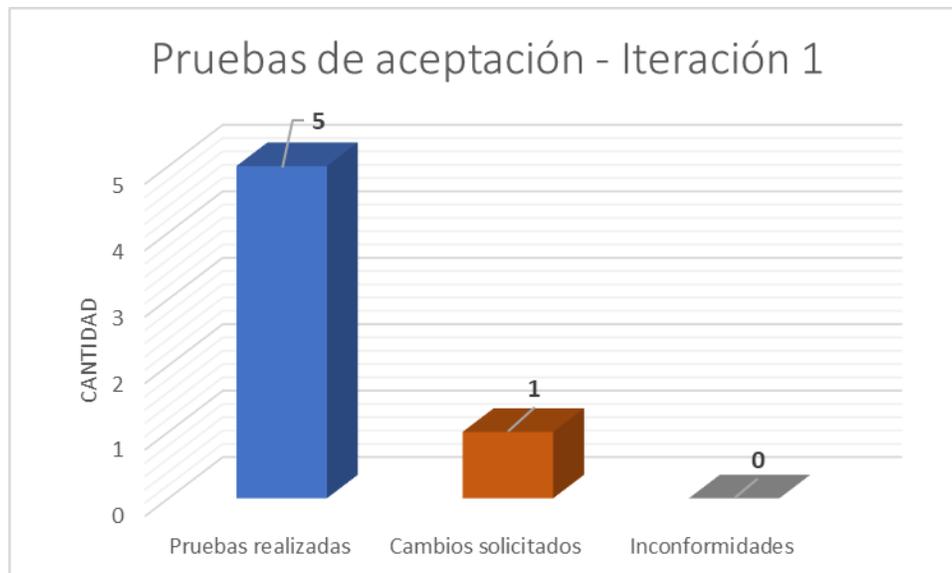


Figura 3.5 Resultados de la ejecución de las pruebas de aceptación de la iteración # 1.

En la iteración 1 se realizaron cinco pruebas de aceptación, una por cada HU las cuales cumplen con los criterios de aceptación del cliente. En la HU Gestionar cliente se solicitó un cambio. Dicho cambio consistió en habilitar la opción de agregar una nueva ciudadanía y un nuevo objeto de arrendamiento.

### Iteración 2

Durante esta iteración se implementaron las HU: Gestionar gastos por arrendamiento de habitaciones, Gestionar gastos por arrendamiento de espacios, Listar gastos de salario, Listar gastos totales, Calcular la disponibilidad de arrendamiento por habitaciones, calcular la disponibilidad de arrendamiento por camas. Para su implementación las mismas fueron divididas en TI. Véase en el Anexo 3.

### Test Unitarios

A continuación, en la figura 3.6 se muestra el resultado de los test unitarios de la iteración #2

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\DayiFQ\Desktop\tesis\conarren> py .\manage.py test
Found 7 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.....
-----
Ran 7 tests in 0.048s

OK
Destroying test database for alias 'default'...
PS C:\Users\DayiFQ\Desktop\tesis\conarren> █
```

Figura 3.6 Resultado de la ejecución de los test unitarios de la iteración #2.

### Pruebas de aceptación

Véase las pruebas de aceptación en el Anexo 7.

A continuación, en la figura 3.7 se muestran los resultados de la ejecución de las pruebas de aceptación de la iteración #2

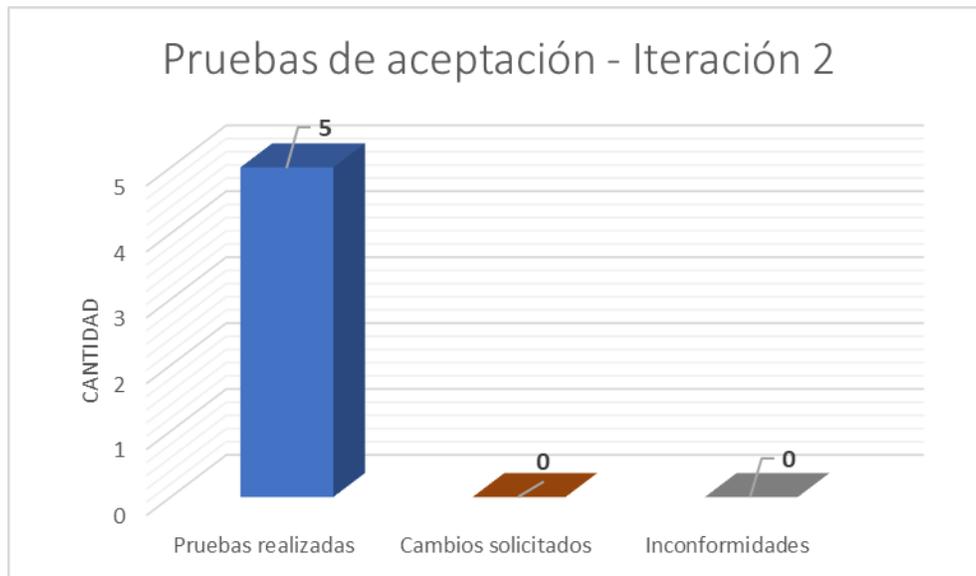


Figura 3.7 Resultados de la ejecución de las pruebas de aceptación de la iteración #2.

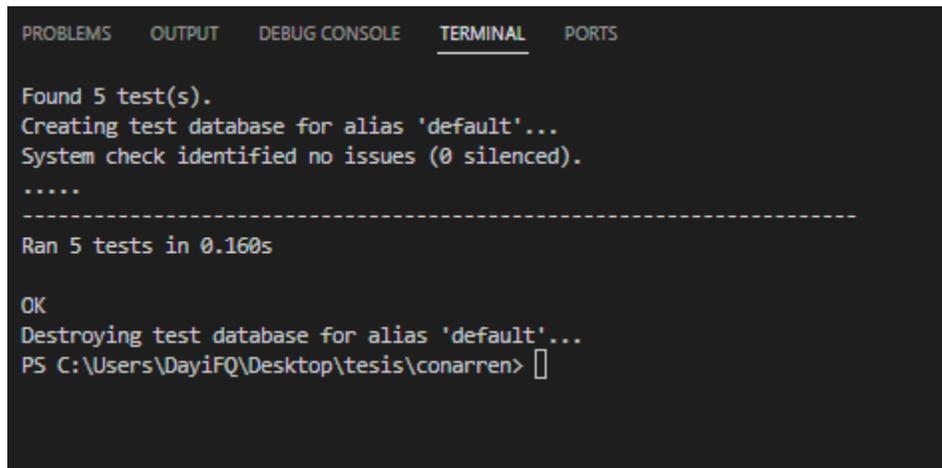
En la iteración 2 se realizaron cinco pruebas de aceptación, una por cada HU las cuales cumplen con los criterios de aceptación del cliente. No fue solicitado ningún cambio ni hubo inconformidades.

### Iteración 3

Durante esta iteración se implementaron las HU: Cumplimentar libros de ingresos y gastos por el arrendamiento de espacios y habitaciones, Calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones y Calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios. Para su implementación las mismas fueron divididas en TI. Véase en el Anexo 4.

### Test Unitarios

A continuación, en la figura 3.8 se muestra el resultado de los test unitarios de la iteración #3



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Found 5 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.....
-----
Ran 5 tests in 0.160s

OK
Destroying test database for alias 'default'...
PS C:\Users\DayiFQ\Desktop\tesis\conarren>
```

Figura 3.8 Resultado de la ejecución de los test unitarios de la iteración #3.

### Pruebas de aceptación

Véase las pruebas de aceptación en el Anexo 8.

A continuación, en la figura 3.9 se muestran los resultados de la ejecución de las pruebas de aceptación de la iteración #3

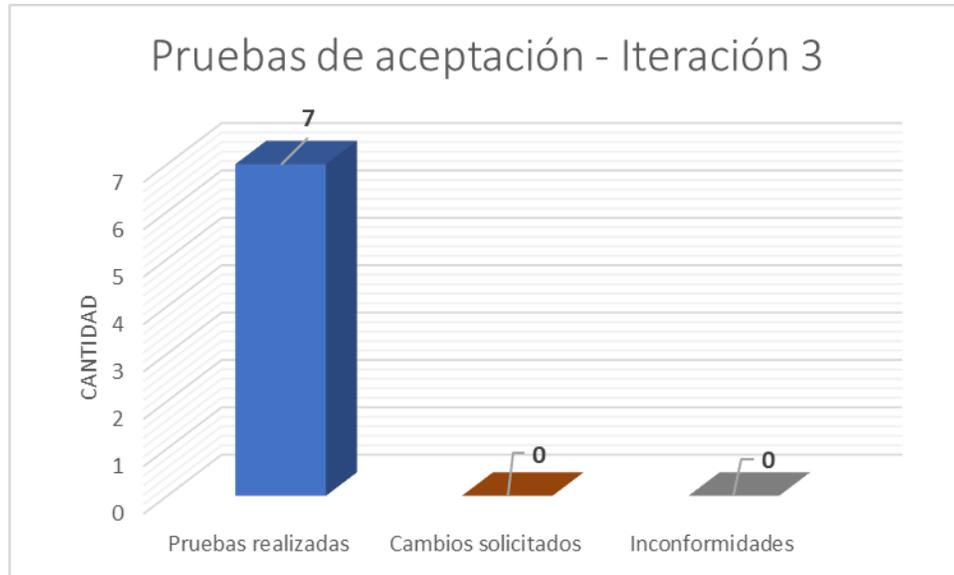


Figura 3.9 Resultados de la ejecución de las pruebas de aceptación de la iteración #3.

En la iteración 3 se realizaron siete pruebas de aceptación, una por cada HU las cuales cumplen con los criterios de aceptación del cliente. No fue solicitado ningún cambio ni hubo inconformidades.

#### **Iteración 4**

Durante esta iteración se implementaron las HU: Cumplimentar vector fiscal, Gestionar las bonificaciones por pago electrónico, Gestionar propietario, Gestionar trabajadores contratados, Cumplimentar declaración jurada y Mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día. Para su implementación las mismas fueron divididas en TI. Véase en el Anexo 5.

#### **Test Unitarios**

A continuación, en la figura 3.10 se muestra el resultado de los test unitarios de la iteración #3

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

System check identified no issues (0 silenced).
..
-----
Ran 2 tests in 0.469s

OK
Destroying test database for alias 'default'...
PS C:\Users\DayiFQ\Desktop\tesis\conarren> █
```

Figura 3.10 Resultado de la ejecución de los test unitarios de la iteración #3.

### Pruebas de aceptación

Véase las pruebas de aceptación en el Anexo 9.

A continuación, en la figura 3.11 se muestran los resultados de la ejecución de las pruebas de aceptación de la iteración #4

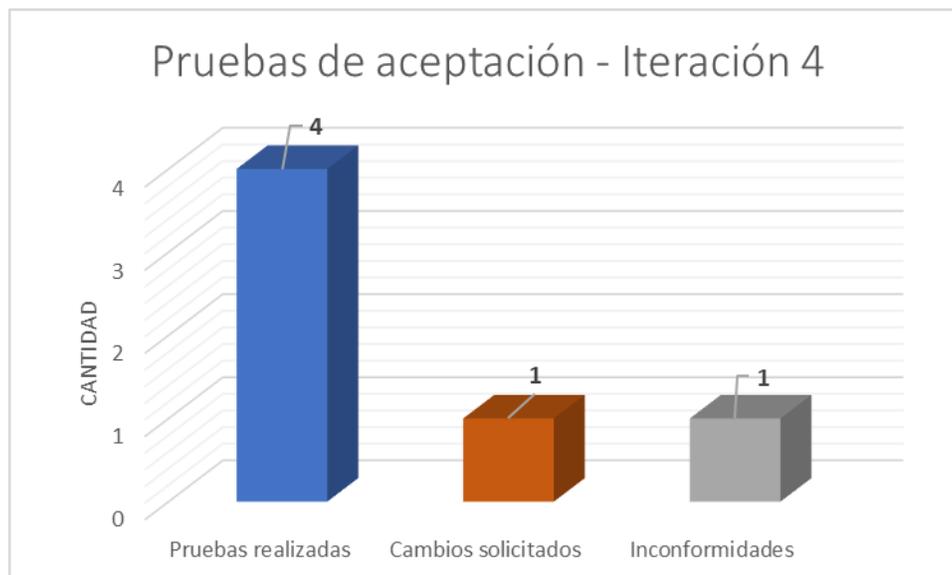


Figura 3.11 Resultados de la ejecución de las pruebas de aceptación de la iteración #4.

En la iteración 4 se realizaron cuatro pruebas de aceptación, una por cada HU las cuales cumplen con los criterios de aceptación del cliente. En la HU gestionar propietario hubo una inconformidad ya que no estaba incluido en el formulario de agregar propietario la fecha de emisión de la declaración jurada. Además, fue solicitado un cambio relacionado con el color

de la tarjeta que mostraba la cantidad de clientes en la HU Mostrar cantidad de clientes que finalizan su estancia.

### **Conclusiones parciales**

Lo anterior lleva a deducir el correcto funcionamiento del sistema y la conformidad del cliente con el resultado luego de ser realizadas las tareas de ingeniería, los test unitarios y las pruebas de aceptación.

## **CONCLUSIONES GENERALES**

Concluyendo, la elaboración y análisis del marco teórico sobre el proceso de cálculo de las obligaciones fiscales de los TCP dedicados al arrendamiento de habitaciones y espacios, así como el estudio de sistemas homólogos demostraron la necesidad de implementar una aplicación que resolviera la problemática existente al inicio de la investigación.

En virtud de lo estudiado, ahora se cuenta con un sistema informático que, a partir del registro de ingresos y gastos, cumplimenta automáticamente los documentos establecidos en la legislación vigente para el cálculo de las obligaciones fiscales.

## **RECOMENDACIONES**

1. Implementar la funcionalidad de generar los modelos submayor, mayor y balance general a partir del registro de los asientos contables.
2. Implementar la funcionalidad de importar la información existente de los arrendamientos.

## REFERENCIAS BIBLIOGRÁFICAS

- Agencia Tributaria: *Suministro Inmediato de Información (SII)*. (2023). Retrieved May 23, 2023, from <https://sede.agenciatributaria.gob.es/Sede/iva/suministro-inmediato-informacion.html>
- Alexandra, A. G. J., & Santiago, T. L. E. (2019). *Desarrollo e implementación de un sistema informático para la gestión del departamento de consejería estudiantil de la unidad educativa "Manuela Cañizares"*.
- Ariel Hernández Cejas, Y. M. P. (n.d.). *Sistema de gestión integral de la planificación y control del servicio de comedores UCI: análisis y diseño del módulo "Gestión de Comensales"*. Universidad de las Ciencias Informáticas.
- Bermúdez, Y. D. (2018, November 19). *Versat Sarasola- Un sistema seguro al alcance de la empresa cubana*. Revista TINO. <https://revista.jovenclub.cu/versat-sarasola-un-sistema-seguro-al-alcance-de-la-empresa-cubana/>
- Capítulo 1: *Descripción general de la arquitectura—Documentación de Odoo 16.0*. (2023). Retrieved November 1, 2023, from [https://www.odoo.com/documentation/16.0/developer/tutorials/getting\\_started/01\\_architecture.html](https://www.odoo.com/documentation/16.0/developer/tutorials/getting_started/01_architecture.html)
- Citmatel. (2020, December 18). *Portal del ciudadano*. <https://www.redisla.gob.cu/es/tramites-y-servicios/servicios/citmatel>
- Contributors, M. O., Jacob Thornton, and Bootstrap. (2023). *Get started with Bootstrap*. Retrieved May 31, 2023, from <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- Desarrollador—*Documentación de Odoo 16.0*. (2023). Retrieved November 1, 2023, from <https://www.odoo.com/documentation/16.0/developer.html>
- Django. (2023). Django Project. Retrieved May 31, 2023, from <https://docs.djangoproject.com/en/4.2/>
- Ganancia. (2023). Retrieved May 23, 2023, from <https://www.nts.go.kr/nts/main.do>
- Herz, J. (2018). *Apuntes de contabilidad financiera* (Universidad Peruana de Ciencias Aplicadas - UPC, Ed.; Tercera). Editorial UPC. <https://doi.org/10.19083/978-612-318-130-7>

Hidalgo de los Santos, V. (2016). *El modelo económico cubano: Reflexiones desde una perspectiva macroeconómica*. *Economía y Desarrollo*, 156(1), 108–130.

Home—Central Board of Direct Taxes, Government of India. (n.d.). Retrieved May 23, 2023, from <https://incometaxindia.gov.in/Pages/default.aspx>

HTML 5: *Qué es, Para qué sirve y Cómo funciona*. (2021, June 18). <https://i.workana.com/glosario/que-es-html-5/>

Ideal Modeling & Diagramming Tool for Agile Team Collaboration. (n.d.). Retrieved May 31, 2023, from <https://www.visual-paradigm.com/>

Ley 113 del Sistema Tributario, *de 23 de julio del 2012*. (n.d.).

López, N. R. (2012). *La aplicación de los impuestos en Cuba*.

Odoo Enterprise vs Community | Odoo Editions Comparison. (n.d.). Odoo S.A. Retrieved November 1, 2023, from <https://www.odoo.com/page/editions>

OMG | Object Management Group. Retrieved from [https://themeforest.net/user/dan\\_fisher](https://themeforest.net/user/dan_fisher). (n.d.). September 30, 2023, from <http://www.omg.org/>

pgAdmin—PostgreSQL Tools. (n.d.). Retrieved May 31, 2023, from <https://www.pgadmin.org/>

Pressman—*Ingeniería del Software. 7ma. Ed. docx*. (n.d.).

Salazar, D. B. (2023). *Tratamiento tributario en el sector educativo: Caso práctico Instituto Tecnológico*. Quito – Ecuador.

Visual Studio: IDE y Editor de código para desarrolladores de software y Teams. (n.d.). Visual Studio. Retrieved May 31, 2023, from <https://visualstudio.microsoft.com/es/>

Welcome to Python.org. (n.d.). Retrieved May 31, 2023, from <https://www.python.org/>

Zea Ordóñez, M. P., Molina Ríos, J. R., & Redrován Castillo, F. F. (2017). *ADMINISTRACIÓN DE BASES DE DATOS CON POSTGRESQL* (1st ed.). Editorial Científica 3Ciencias. <https://doi.org/10.17993/IngyTec.2017.18>

# ANEXOS

## Anexo 1

### A.1. Historias de Usuarios

Tabla A1.30 Historia de Usuario #3.

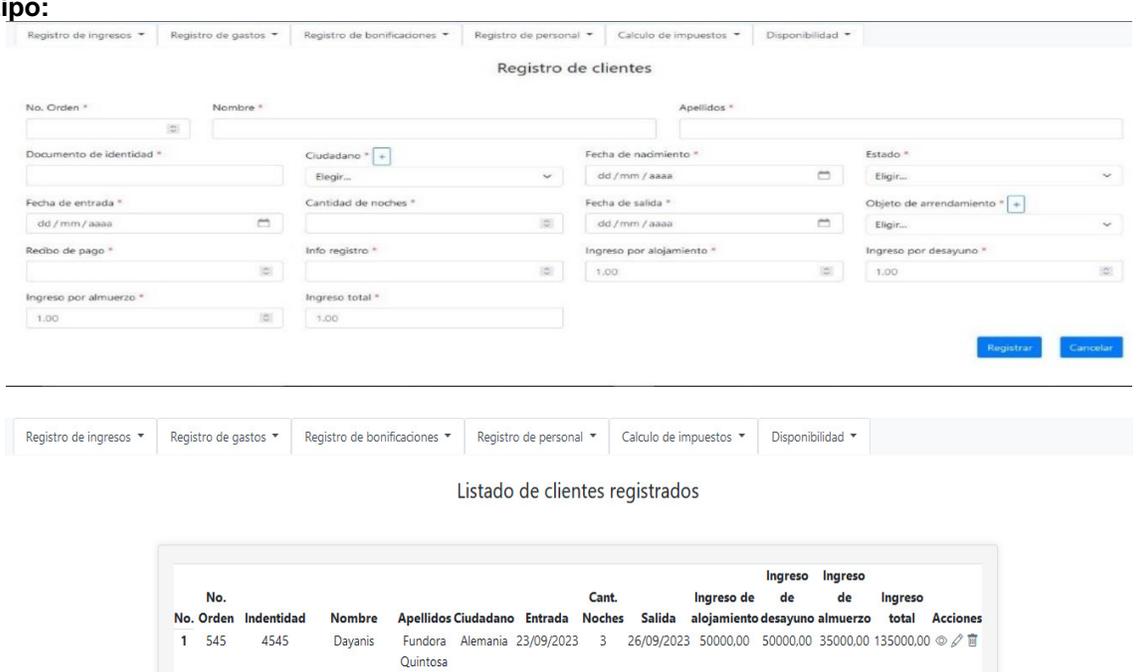
Historia de Usuario																													
<b>Número: 3</b>	<b>Nombre: Gestionar cliente</b>																												
<b>Usuario: Usuario</b>																													
<b>Prioridad en el negocio: Alta</b>	<b>Riesgo en desarrollo: Alto</b>																												
<b>Puntos estimados: 1.1</b>	<b>Iteración asignada: 1</b>																												
<b>Programador responsable: Dayanis Fundora Quintosa</b>																													
<p><b>Descripción:</b> Ofrece al usuario una vista para llenar un formulario con los siguientes datos: No. de orden, documento de identidad, nombre, apellidos, ciudadano, fecha de nacimiento, estado, fecha de entrada, fecha de salida, cantidad de noches, objeto de arrendamiento, recibo de pago, info. Registro, ingreso de arrendamiento, ingreso de desayuno, ingreso de almuerzo, ingreso total. El dato fecha de salida debe cumplimentarse automáticamente por el sistema, al ser insertada la fecha de entrada y la cantidad de noches por el usuario (fecha de salida = fecha de llegada + cantidad de noches). También permite listar, ver, modificar y eliminar los clientes registrados. En los casos de los datos de ciudadanía y objeto de arrendamiento debe dar la posibilidad de agregar un nuevo dato, modificarlo o eliminarlo. Además, se deben listar los clientes registrados, así como los clientes que se encuentran hospedados en la fecha en que se emita el listado.</p>																													
<p><b>Observaciones:</b> Estos datos deben quedar registrados en una base de datos para su posterior consulta, modificación, eliminación en caso de que sea necesario. Además, el usuario debe tener la opción de cancelar el registro de ser necesario.</p>																													
<p><b>Prototipo:</b></p>  <p>The screenshot displays a web interface for client registration. At the top, there are navigation tabs: 'Registro de ingresos', 'Registro de gastos', 'Registro de bonificaciones', 'Registro de personal', 'Calculo de impuestos', and 'Disponibilidad'. The main section is titled 'Registro de clientes' and contains a form with the following fields: 'No. Orden *', 'Nombre *', 'Apellidos *', 'Documento de identidad *', 'Ciudadano *' (with a dropdown menu), 'Fecha de nacimiento *' (dd/mm/aaaa), 'Estado *' (with a dropdown menu), 'Fecha de entrada *' (dd/mm/aaaa), 'Cantidad de noches *', 'Fecha de salida *' (dd/mm/aaaa), 'Objeto de arrendamiento *' (with a dropdown menu), 'Recibo de pago *', 'Info registro *', 'Ingreso por alojamiento *' (1.00), 'Ingreso por desayuno *' (1.00), and 'Ingreso por almuerzo *' (1.00). There is an 'Ingreso total *' field showing 1.00. At the bottom right of the form are 'Registrar' and 'Cancelar' buttons.</p> <p>Below the form is a section titled 'Listado de clientes registrados' which contains a table with the following data:</p> <table border="1"> <thead> <tr> <th>No.</th> <th>No. Orden</th> <th>Indentidad</th> <th>Nombre</th> <th>Apellidos</th> <th>Ciudadano</th> <th>Entrada</th> <th>Noches</th> <th>Salida</th> <th>Ingreso de alojamiento</th> <th>Ingreso de desayuno</th> <th>Ingreso de almuerzo</th> <th>Ingreso total</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>545</td> <td>4545</td> <td>Dayanis</td> <td>Fundora</td> <td>Alemania</td> <td>23/09/2023</td> <td>3</td> <td>26/09/2023</td> <td>50000.00</td> <td>50000.00</td> <td>35000.00</td> <td>135000.00</td> <td>🔍 🗑️</td> </tr> </tbody> </table>		No.	No. Orden	Indentidad	Nombre	Apellidos	Ciudadano	Entrada	Noches	Salida	Ingreso de alojamiento	Ingreso de desayuno	Ingreso de almuerzo	Ingreso total	Acciones	1	545	4545	Dayanis	Fundora	Alemania	23/09/2023	3	26/09/2023	50000.00	50000.00	35000.00	135000.00	🔍 🗑️
No.	No. Orden	Indentidad	Nombre	Apellidos	Ciudadano	Entrada	Noches	Salida	Ingreso de alojamiento	Ingreso de desayuno	Ingreso de almuerzo	Ingreso total	Acciones																
1	545	4545	Dayanis	Fundora	Alemania	23/09/2023	3	26/09/2023	50000.00	50000.00	35000.00	135000.00	🔍 🗑️																

Tabla A1.31 Historia de Usuario #4.

Historia de Usuario													
<b>Número:</b> 4	<b>Nombre:</b> Gestionar el ingreso por arrendamiento de espacios												
<b>Usuario:</b> Usuario													
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto												
<b>Puntos estimados:</b> 0.9	<b>Iteración asignada:</b> 1												
<b>Programador responsable:</b> Dayanis Fundora Quintosa													
<b>Descripción:</b> El usuario debe llenar un formulario con los siguientes datos: fecha, importe cobrado, cantidad de espacios, período cobrado que puede ser el nombre del mes, fecha de ingreso. Tiene que haber un botón que permita registrar y otro cancelar. También permite listar, ver, modificar y eliminar los espacios registrados.													
<b>Observaciones:</b> Ninguna													
<b>Prototipo:</b>													
<p>The screenshot displays a web interface for recording rental income. At the top, there is a navigation bar with dropdown menus for 'Registro de ingresos', 'Registro de gastos', 'Registro de bonificaciones', 'Registro de personal', 'Calculo de impuestos', and 'Disponibilidad'. Below this, the main heading is 'Registro del ingreso de arrendamiento de espacios'. The form contains several input fields: 'Cantidad de espacios arrendados *' (with a value of 6), 'Importe cobrado *' (with a value of 36800.00), 'Periodo cobrado *' (a dropdown menu showing 'Elegir...'), and 'Fecha de ingreso *' (a date picker showing 'dd/mm/aaaa'). There are two buttons, 'Registrar' and 'Cancelar', below the form. Below the form, there is a section titled 'Listado de ingresos de espacio' which contains a table with the following data:</p> <table border="1"> <thead> <tr> <th>No.</th> <th>Cantidad espacio</th> <th>Importe cobrado</th> <th>Periodo cobrado</th> <th>Fecha</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>6</td> <td>36800.00</td> <td>Febrero</td> <td>02/02/2023</td> <td> </td> </tr> </tbody> </table>		No.	Cantidad espacio	Importe cobrado	Periodo cobrado	Fecha	Acciones	1	6	36800.00	Febrero	02/02/2023	
No.	Cantidad espacio	Importe cobrado	Periodo cobrado	Fecha	Acciones								
1	6	36800.00	Febrero	02/02/2023									

Tabla A1.32 Historia de Usuario #5.

Historia de Usuario	
<b>Número:</b> 5	<b>Nombre:</b> Gestionar las bonificaciones por pago electrónico
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 0.6	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Debe aparecer un formulario para que el usuario introduzca la fecha, el mes, y la cifra bonificada y tener dos botones, uno para registrar y otro para cancelar. Además, debe aparecer una tabla en la cual se van listando las bonificaciones registradas con la opción de editar y eliminar.	
<b>Observaciones:</b> Ninguna	
<b>Prototipo:</b>	

Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾

Bonificaciones recibidas por el pago de tributos realizados por canales electrónicos

Fecha:  
 Mes: 
 Cifra bonificada:

No.	Fecha	Mes	Cifra bonificada	Acciones
1	08/04/2023	Abril	360,0	

Tabla A1.33 Historia de Usuario #6.

Historia de Usuario																			
<b>Número: 6</b>	<b>Nombre:</b> Gestionar los gastos por el arrendamiento de espacios																		
<b>Usuario:</b> Usuario																			
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto																		
<b>Puntos estimados:</b> 1.1	<b>Iteración asignada:</b> 2																		
<b>Programador responsable:</b> y Dayanis Fundora Quintosa																			
<p><b>Descripción:</b> En la opción “Gastos por el arrendamiento de espacios” el usuario debe llenar un formulario con los siguientes datos: fecha, detalle del gasto, concepto, unidad de medida, cantidad, precio unitario, importe, importe pagado, bonificación por pago electrónico. En el dato concepto de gastos el usuario debe agregar otro si es necesario. Debe haber dos botones uno para registrar y otro para cancelar. También permite listar, ver, modificar y eliminar los gastos registrados.</p> <p><b>Observaciones:</b> Estos datos deben quedar registrados en una base de datos para su posterior consulta, modificación, eliminación en caso de que sea necesario. Además, el usuario debe tener la opción de cancelar el registro de ser necesario.</p>																			
<p><b>Prototipo:</b></p> <p>             Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾           </p> <p style="text-align: center;">Registro de gastos por el arrendamiento de espacios</p> <p>             Fecha: <input type="text" value="dd/mm/aaaa"/> <input type="calendar"/>             Detalle del gasto: <input type="text"/> </p> <p>             Concepto: <input type="text" value="Elegir..."/>             Unidad de medida: <input type="text" value="Elegir..."/>             Cantidad: <input type="text" value="1.00"/>             Precio unitario: <input type="text" value="1.00"/> </p> <p>             Importe: <input type="text" value="1.00"/>             Importe Pagado: <input type="text" value="1.00"/>             Bonificación por pago electrónico: <input type="text" value="1.00"/> </p> <p style="text-align: center;"> <input type="button" value="Registrar"/> <input type="button" value="Cancelar"/> </p> <p>             Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾           </p> <p style="text-align: center;">Listado de gastos por el arrendamiento de espacio</p> <table border="1"> <thead> <tr> <th>No.</th> <th>Fecha</th> <th>Detalle del gasto</th> <th>Concepto</th> <th>UM</th> <th>Cantidad</th> <th>Precio Unitario</th> <th>Importe</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>10/08/2023</td> <td>cvc</td> <td>Electricidad</td> <td>Ninguna</td> <td>3000</td> <td>3,00</td> <td>9000,00</td> <td> </td> </tr> </tbody> </table>		No.	Fecha	Detalle del gasto	Concepto	UM	Cantidad	Precio Unitario	Importe	Acciones	1	10/08/2023	cvc	Electricidad	Ninguna	3000	3,00	9000,00	
No.	Fecha	Detalle del gasto	Concepto	UM	Cantidad	Precio Unitario	Importe	Acciones											
1	10/08/2023	cvc	Electricidad	Ninguna	3000	3,00	9000,00												

Tabla A1.34 Historia de Usuario #7.

Historia de Usuario																																					
<b>Número:</b> 7	<b>Nombre:</b> Gestionar gastos por arrendamiento de habitaciones																																				
<b>Usuario:</b> Usuario																																					
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto																																				
<b>Puntos estimados:</b> 1.1	<b>Iteración asignada:</b> 2																																				
<b>Programador responsable:</b> Dayanis Fundora Quintosa																																					
<p><b>Descripción:</b> En la opción “Gastos por el arrendamiento de habitaciones” el usuario debe llenar un formulario con los siguientes datos: fecha, detalle del gasto, concepto, unidad de medida, cantidad, precio unitario, importe, importe pagado, bonificación por pago electrónico. En el dato concepto de gastos el usuario debe tener la opción de elegir entre los conceptos: electricidad, teléfono, agua, gas, salario, almuerzo, desayuno, limpieza, avituallamiento, documentos, mantenimiento de electricidad o plomería, así como agregar otro si es necesario. En el caso de la unidad de medida también debe tener la opción de agregar una nueva. Debe haber dos botones uno para registrar y otro para cancelar. También debe tener la opción de listar, ver, modificar y eliminar los gastos registrados.</p>																																					
<p><b>Observaciones:</b> Estos datos deben quedar registrados en una base de datos para su posterior consulta, modificación, eliminación en caso de que sea necesario. Además, el usuario debe tener la opción de cancelar el registro de ser necesario.</p>																																					
<p><b>Prototipo:</b></p> <p>Registro de gastos por el arrendamiento de habitaciones</p> <p>Listado de gastos por el arrendamiento de habitaciones</p> <table border="1"> <thead> <tr> <th>No.</th> <th>Fecha</th> <th>Detalle del gasto</th> <th>Concepto</th> <th>UM</th> <th>Cantidad</th> <th>Precio Unitario</th> <th>Importe</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>26/08/2023</td> <td>Electricidad Liset</td> <td>Electricidad</td> <td>Ninguna</td> <td>1</td> <td>100,00</td> <td>100,00</td> <td>🔍 ✎ 🗑️</td> </tr> <tr> <td>2</td> <td>07/09/2023</td> <td>dfgfdg</td> <td>Electricidad</td> <td>Ninguna</td> <td>5</td> <td>5,00</td> <td>25,00</td> <td>🔍 ✎ 🗑️</td> </tr> <tr> <td>3</td> <td>13/09/2023</td> <td>Documento</td> <td>Documento</td> <td>Ninguna</td> <td>5</td> <td>500,00</td> <td>2500,00</td> <td>🔍 ✎ 🗑️</td> </tr> </tbody> </table>		No.	Fecha	Detalle del gasto	Concepto	UM	Cantidad	Precio Unitario	Importe	Acciones	1	26/08/2023	Electricidad Liset	Electricidad	Ninguna	1	100,00	100,00	🔍 ✎ 🗑️	2	07/09/2023	dfgfdg	Electricidad	Ninguna	5	5,00	25,00	🔍 ✎ 🗑️	3	13/09/2023	Documento	Documento	Ninguna	5	500,00	2500,00	🔍 ✎ 🗑️
No.	Fecha	Detalle del gasto	Concepto	UM	Cantidad	Precio Unitario	Importe	Acciones																													
1	26/08/2023	Electricidad Liset	Electricidad	Ninguna	1	100,00	100,00	🔍 ✎ 🗑️																													
2	07/09/2023	dfgfdg	Electricidad	Ninguna	5	5,00	25,00	🔍 ✎ 🗑️																													
3	13/09/2023	Documento	Documento	Ninguna	5	500,00	2500,00	🔍 ✎ 🗑️																													

Tabla A1.35 Historia de Usuario #8.

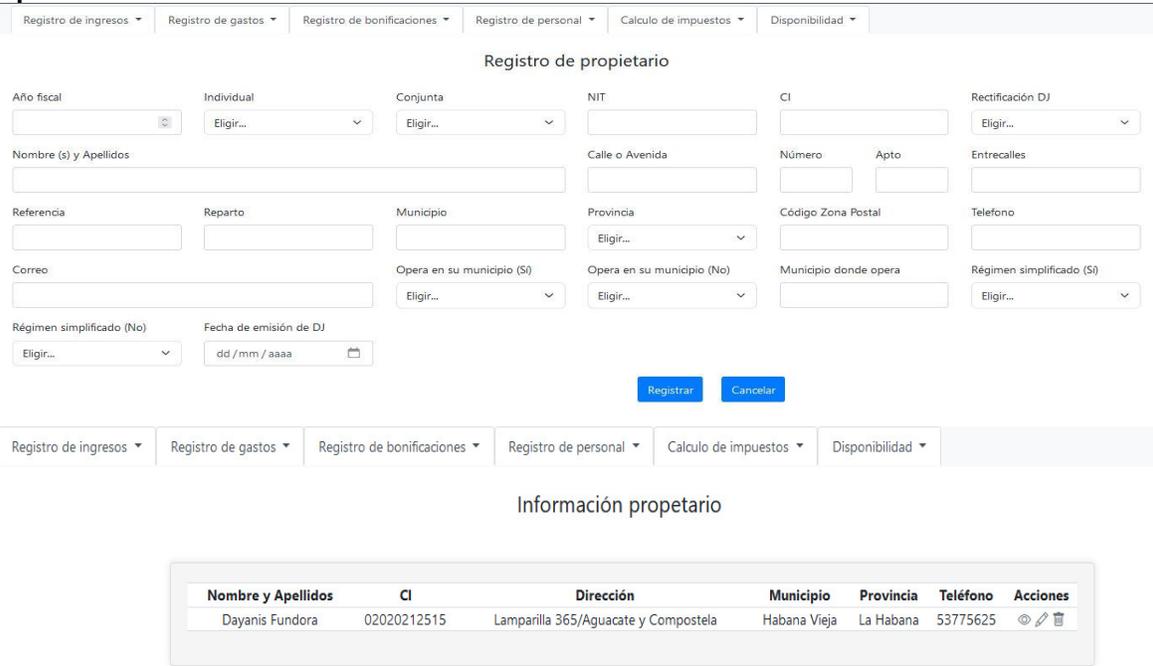
Historia de Usuario															
<b>Número: 8</b>	<b>Nombre: Gestionar propietario</b>														
<b>Usuario: Usuario</b>															
<b>Prioridad en el negocio: Alta</b>	<b>Riesgo en desarrollo: Alto</b>														
<b>Puntos estimados: 0.6</b>	<b>Iteración asignada: 2</b>														
<b>Programador responsable: Dayanis Fundora Quintosa</b>															
<p><b>Descripción:</b> Debe aparecer un formulario para que el usuario introduzca los datos: año fiscal, individual, conjunta, NIT, CI, Rectificación de la DJ, nombre, apellidos, dirección completa, código postal, teléfono, correo, si opera en su municipio o no, municipio donde opera, régimen simplificado, fecha de emisión de la DJ y tener dos botones, uno para registrar y otro para cancelar. También debe tener la opción de listar, ver, modificar y eliminar el propietario registrado.</p>															
<p><b>Observaciones:</b> Con los datos que introduce el usuario se debe cumplimentar la sección 1 de la Declaración Jurada.</p>															
<p><b>Prototipo:</b></p>  <p>The screenshot shows a web interface with a top navigation bar containing menu items like 'Registro de ingresos', 'Registro de gastos', etc. The main content area is titled 'Registro de propietario' and contains a form with various input fields: 'Año fiscal', 'Individual' (radio button), 'Conjunta' (radio button), 'NIT', 'CI', 'Rectificación DJ', 'Nombre (s) y Apellidos', 'Calle o Avenida', 'Número', 'Apto', 'Entrecalles', 'Referencia', 'Reparto', 'Municipio', 'Provincia', 'Código Zona Postal', 'Telefono', 'Correo', 'Opera en su municipio (Si)', 'Opera en su municipio (No)', 'Municipio donde opera', 'Régimen simplificado (Si)', 'Régimen simplificado (No)', and 'Fecha de emisión de DJ'. There are 'Registrar' and 'Cancelar' buttons. Below the form is a table titled 'Información propietario' with the following data:</p> <table border="1"> <thead> <tr> <th>Nombre y Apellidos</th> <th>CI</th> <th>Dirección</th> <th>Municipio</th> <th>Provincia</th> <th>Teléfono</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>Dayanis Fundora</td> <td>02020212515</td> <td>Lamparilla 365/Aguacate y Compostela</td> <td>Habana Vieja</td> <td>La Habana</td> <td>53775625</td> <td>  </td> </tr> </tbody> </table>		Nombre y Apellidos	CI	Dirección	Municipio	Provincia	Teléfono	Acciones	Dayanis Fundora	02020212515	Lamparilla 365/Aguacate y Compostela	Habana Vieja	La Habana	53775625	  
Nombre y Apellidos	CI	Dirección	Municipio	Provincia	Teléfono	Acciones									
Dayanis Fundora	02020212515	Lamparilla 365/Aguacate y Compostela	Habana Vieja	La Habana	53775625	  									

Tabla A1.36 Historia de Usuario #9.

Historia de Usuario	
<b>Número: 9</b>	<b>Nombre: Gestionar trabajadores contratados</b>
<b>Usuario: Usuario</b>	
<b>Prioridad en el negocio: Alta</b>	<b>Riesgo en desarrollo: Alto</b>
<b>Puntos estimados: 0.6</b>	<b>Iteración asignada: 2</b>
<b>Programador responsable: Dayanis Fundora Quintosa</b>	

**Descripción:** Debe aparecer un formulario para que el usuario introduzca los datos: código de actividad, nombre y apellidos, período de contratación, municipio de residencia, CI, importe de salario. Además, debe tener dos botones, uno para registrar y otro para cancelar. También debe tener la opción de listar, ver, modificar y eliminar los trabajadores registrados.

**Observaciones:** Con los datos que introduce el usuario se debe cumplimentar la sección I de la Declaración Jurada.

**Prototipo:**

Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾

Registro de trabajadores contratados

Código actividad  Nombre y Apellidos  Período de contratación desde

Período de contratación hasta  Municipio de residencia  CI  Importe

Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾

Listado de trabajadores contratados

Codigo actividad	Nombre y Apellidos	Contratado desde	Contratado hasta	Municipio	CI	Importe	Acciones
689	Diana Madiedo	10/08/2023	26/08/2023	Habana Vieja	97020108255	3600,00	

Tabla A1.37 Historia de Usuario # 10.

Historia de Usuario									
<b>Número: 10</b>	<b>Nombre:</b> Calcular índice ocupacional								
<b>Usuario:</b> Usuario									
<b>Prioridad en el negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Alto								
<b>Puntos estimados:</b> 0.4	<b>Iteración asignada:</b> 2								
<b>Programador responsable:</b> Dayanis Fundora Quintosa									
<b>Descripción:</b> A partir de la capacidad que tiene el hostel y de la cantidad de clientes registrados se tiene que calcular el índice ocupacional que ha habido mensualmente.									
<b>Observaciones:</b> El resultado tiene que darse en porciento.									
<b>Prototipo:</b>									
<p>Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾</p> <p style="text-align: center;">Índice ocupacional</p> <p style="text-align: center;"><u>Capacidad: 34</u></p> <table border="1"> <thead> <tr> <th>Mes</th> <th>Cantidad de días del mes</th> <th>Cantidad de noches</th> <th>Índice ocupacional</th> </tr> </thead> <tbody> <tr> <td>Enero</td> <td>31</td> <td>0</td> <td>0,00%</td> </tr> </tbody> </table>		Mes	Cantidad de días del mes	Cantidad de noches	Índice ocupacional	Enero	31	0	0,00%
Mes	Cantidad de días del mes	Cantidad de noches	Índice ocupacional						
Enero	31	0	0,00%						

Tabla A1.38 Historia de Usuario # 11.

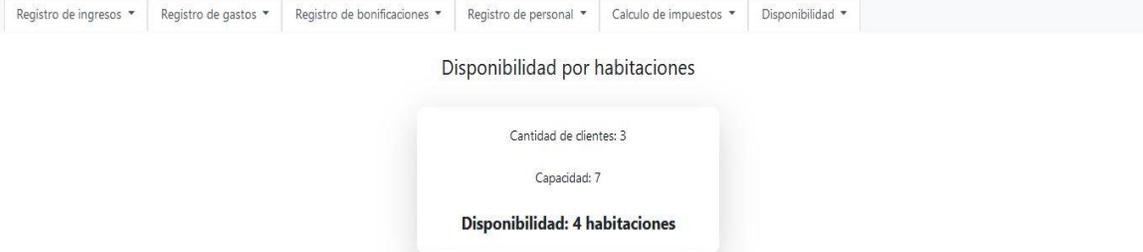
Historia de Usuario	
<b>Número:</b> 11	<b>Nombre:</b> Calcular la disponibilidad de arrendamiento por habitaciones
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Bajo	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 0.3	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Dayanis Fundora Quintosa	
<b>Descripción:</b> El usuario en esta opción debe ver en pantalla la cantidad de clientes hospedados, la capacidad del hostel por habitaciones y la disponibilidad existente. Dicha disponibilidad será calculada automáticamente por el sistema.	
<b>Observaciones:</b> En esta opción de arrendamiento por habitación es requisito indispensable que el sistema cuente solamente los clientes que fueron registrados como principales y que además aún están en la casa ya que en una habitación puede haber más de un cliente y el dato "Principal" es lo que define que ocupa una habitación o más de una.	
<b>Prototipo:</b>	
	

Tabla A1.39 Historia de Usuario # 12.

Historia de Usuario	
<b>Número:</b> 12	<b>Nombre:</b> Calcular la disponibilidad de arrendamiento por camas
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Bajo	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 0.3	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Dayanis Fundora Quintosa	
<b>Descripción:</b> El usuario en esta opción debe ver en pantalla la cantidad de clientes hospedados, la capacidad del hostel por camas y la disponibilidad existente. Dicha disponibilidad será calculada automáticamente por el sistema.	
<b>Observaciones:</b> En esta opción la disponibilidad se obtiene restando la capacidad por camas del hostel con la cantidad de clientes registrados que aún están hospedados.	
<b>Prototipo:</b>	

Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾

Disponibilidad por camas

Cantidad de clientes: 3

Capacidad: 34

**Disponibilidad: 31 camas**

Tabla A1.40 Historia de Usuario # 13.

Historia de Usuario																																		
<b>Número: 13</b>	<b>Nombre:</b> Calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones																																	
<b>Usuario:</b> Usuario																																		
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto																																	
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3																																	
<b>Programador responsable:</b> Dayanis Fundora Quintosa																																		
<p><b>Descripción:</b> Debe aparecer una tabla con los datos establecidos por la ONAT: mes, impuesto sobre las ventas o servicios, impuestos por la utilización de la fuerza de trabajo, impuesto sobre documentos, tasa por radicación de anuncios, contribución a la seguridad social, otros, subtotal, restauración, cuota, pago. Cada fila debe tener cumplimentados los datos mensuales. En la columna “Impuestos sobre las ventas o servicios” el dato que se debe cumplimentar es el 15% del total de ingresos mensuales por el arrendamiento de habitaciones. En la columna “Utilización de la fuerza de trabajo” se debe cumplimentar el dato si hay trabajadores contratados y para la programación de este dato se debe ver la escala que establece la ONAT en el vector fiscal. En la columna “Impuestos sobre documentos” se debe reflejar el gasto en el que se haya incurrido por compra de documentos según los gastos registrados por el usuario. En la columna “Contribución a la seguridad social” se debe cumplimentar el dato que aparece en el vector fiscal correspondientes a dicho pago. En la columna “Subtotal” deben sumarse los datos de la fila. En la columna “Restauración” se debe reflejar el 10% del 5% del total de ingresos por el arrendamiento de habitaciones. Debe totalizarse todo por columnas para obtener los totales anuales de cada concepto que deben cumplimentarse en la Declaración Jurada.</p>																																		
<b>Observaciones:</b> Todos los datos de la tabla deben ser cumplimentados automáticamente.																																		
<b>Prototipo:</b>																																		
<p>             Registro de ingresos ▾ Registro de gastos ▾ Registro de bonificaciones ▾ Registro de personal ▾ Calculo de impuestos ▾ Disponibilidad ▾           </p> <p style="text-align: center;">Pago de tributos por el arrendamiento de habitaciones</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="10">Tributos pagados deducibles a la declaración jurada</th> <th>Gastos deducibles de la base imponible</th> </tr> <tr> <th>Mes</th> <th>Impuesto sobre las ventas o los servicios</th> <th>Impuesto por la utilización de la fuerza de trabajo</th> <th>Impuesto sobre documentos</th> <th>Tasa por la radicación de anuncios y propaganda comercial</th> <th>Contribución especial a la seguridad social</th> <th>Otros</th> <th>Subtotal</th> <th>Contribución para la restauración y preservación de la zona donde desarrolla la actividad</th> <th>Cuota mensual</th> <th>Pago por arrendamiento de bienes y actividades estatales</th> </tr> </thead> <tbody> <tr> <td>Enero</td> <td style="text-align: center;">0</td> <td></td> <td style="text-align: center;">0</td> <td></td> <td></td> <td></td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td></td> <td></td> </tr> </tbody> </table>		Tributos pagados deducibles a la declaración jurada										Gastos deducibles de la base imponible	Mes	Impuesto sobre las ventas o los servicios	Impuesto por la utilización de la fuerza de trabajo	Impuesto sobre documentos	Tasa por la radicación de anuncios y propaganda comercial	Contribución especial a la seguridad social	Otros	Subtotal	Contribución para la restauración y preservación de la zona donde desarrolla la actividad	Cuota mensual	Pago por arrendamiento de bienes y actividades estatales	Enero	0		0				0	0		
Tributos pagados deducibles a la declaración jurada										Gastos deducibles de la base imponible																								
Mes	Impuesto sobre las ventas o los servicios	Impuesto por la utilización de la fuerza de trabajo	Impuesto sobre documentos	Tasa por la radicación de anuncios y propaganda comercial	Contribución especial a la seguridad social	Otros	Subtotal	Contribución para la restauración y preservación de la zona donde desarrolla la actividad	Cuota mensual	Pago por arrendamiento de bienes y actividades estatales																								
Enero	0		0				0	0																										

Tabla A1.41 Historia de Usuario # 14.

Historia de Usuario	
<b>Número: 14</b>	<b>Nombre:</b> Calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto

<b>Puntos estimados: 1</b>	<b>Iteración asignada: 3</b>																																		
<b>Programador responsable:</b> Dayanis Fundora Quintosa																																			
<p><b>Descripción:</b> Debe aparecer una tabla con los datos establecidos por la ONAT: mes, impuesto sobre las ventas o servicios, impuestos por la utilización de la fuerza de trabajo, impuesto sobre documentos, tasa por radicación de anuncios, contribución a la seguridad social, otros, subtotal, restauración, cuota, pago. Cada fila debe tener cumplimentados los datos mensuales. En la columna “Impuestos sobre las ventas o servicios” el dato que se debe cumplimentar es el 15% del total de ingresos mensuales por el arrendamiento de espacios. En la columna “Utilización de la fuerza de trabajo” se debe cumplimentar el dato si hay trabajadores contratados y para la programación de este dato se debe ver la escala que establece la ONAT en el vector fiscal. En la columna “Impuestos sobre documentos” se debe reflejar el gasto en el que se haya incurrido por compra de documentos según los gastos registrados por el usuario. En la columna “Contribución a la seguridad social” se debe cumplimentar el dato que aparece en el vector fiscal correspondientes a dicho pago. En la columna “Subtotal” deben sumarse los datos de la fila. En la columna “Restauración” se debe reflejar el 10% del 5% del total de ingresos por el arrendamiento de espacios. Debe totalizarse todo por columnas para obtener los totales anuales de cada concepto que deben cumplimentarse en la Declaración Jurada.</p>																																			
<p><b>Observaciones:</b> Todos los datos de la tabla deben ser cumplimentados automáticamente.</p>																																			
<p><b>Prototipo:</b></p> <p>Registro de ingresos ▼ Registro de gastos ▼ Registro de bonificaciones ▼ Registro de personal ▼ Calculo de impuestos ▼ Disponibilidad ▼</p> <p>Pago de tributos por el arrendamiento de espacios</p> <table border="1"> <thead> <tr> <th colspan="11">Tributos pagados deducibles a la declaración jurada</th> <th>Gastos deducibles de la base imponible</th> </tr> <tr> <th>Mes</th> <th>Impuesto sobre las ventas o los servicios</th> <th>Impuesto por la utilización de la fuerza de trabajo</th> <th>Impuesto sobre documentos</th> <th>Tasa por la radicación de anuncios y propaganda comercial</th> <th>Contribución especial a la seguridad social</th> <th>Otros</th> <th>Subtotal</th> <th>Contribución para la restauración y preservación de la zona donde desarrolla la actividad</th> <th>Cuota mensual</th> <th>Pago por arrendamiento de bienes y actividades estatales</th> </tr> </thead> <tbody> <tr> <td>Enero</td> <td>0</td> <td></td> <td>0</td> <td></td> <td></td> <td></td> <td>0</td> <td>0</td> <td></td> <td></td> </tr> </tbody> </table>		Tributos pagados deducibles a la declaración jurada											Gastos deducibles de la base imponible	Mes	Impuesto sobre las ventas o los servicios	Impuesto por la utilización de la fuerza de trabajo	Impuesto sobre documentos	Tasa por la radicación de anuncios y propaganda comercial	Contribución especial a la seguridad social	Otros	Subtotal	Contribución para la restauración y preservación de la zona donde desarrolla la actividad	Cuota mensual	Pago por arrendamiento de bienes y actividades estatales	Enero	0		0				0	0		
Tributos pagados deducibles a la declaración jurada											Gastos deducibles de la base imponible																								
Mes	Impuesto sobre las ventas o los servicios	Impuesto por la utilización de la fuerza de trabajo	Impuesto sobre documentos	Tasa por la radicación de anuncios y propaganda comercial	Contribución especial a la seguridad social	Otros	Subtotal	Contribución para la restauración y preservación de la zona donde desarrolla la actividad	Cuota mensual	Pago por arrendamiento de bienes y actividades estatales																									
Enero	0		0				0	0																											

Tabla A1.42 Historia de Usuario # 15.

<b>Historia de Usuario</b>									
<b>Número: 15</b>	<b>Nombre:</b> Listar los ingresos totales								
<b>Usuario:</b> Usuario									
<b>Prioridad en el negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Alto								
<b>Puntos estimados: 0.4</b>	<b>Iteración asignada: 3</b>								
<b>Programador responsable:</b> Dayanis Fundora Quintosa									
<p><b>Descripción:</b> Debe aparecer una tabla en la cual el usuario pueda ver los datos: mes, ingreso por arrendamiento, ingreso por espacios, total de ingreso por arrendamiento y espacios con totales mensuales y anuales.</p>									
<p><b>Observaciones:</b> En esta tabla el usuario solo debe ver el total mensual de ingresos de ambas actividades y el anual. Los datos que aparecerían en esta tabla deben tomarse del registro de clientes y del registro de ingreso por el arrendamiento de espacios.</p>									
<p><b>Prototipo:</b></p> <p>Registro de ingresos ▼ Registro de gastos ▼ Registro de bonificaciones ▼ Registro de personal ▼ Calculo de impuestos ▼ Disponibilidad ▼</p> <p>Listado de ingresos totales de arrendamiento y espacio</p> <table border="1"> <thead> <tr> <th>Mes</th> <th>Ingreso por arrendamiento</th> <th>Ingreso por espacio</th> <th>Total de ingreso por arrendamiento y espacio</th> </tr> </thead> <tbody> <tr> <td>Enero</td> <td>0,00</td> <td>0,00</td> <td>0,00</td> </tr> </tbody> </table>		Mes	Ingreso por arrendamiento	Ingreso por espacio	Total de ingreso por arrendamiento y espacio	Enero	0,00	0,00	0,00
Mes	Ingreso por arrendamiento	Ingreso por espacio	Total de ingreso por arrendamiento y espacio						
Enero	0,00	0,00	0,00						

Tabla A1.43 Historia de Usuario # 16.

Historia de Usuario					
<b>Número: 16</b>	<b>Nombre:</b> Listar gastos de salario				
<b>Usuario:</b> Usuario					
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto				
<b>Puntos estimados: 0.4</b>	<b>Iteración asignada: 3</b>				
<b>Programador responsable:</b> Dayanis Fundora Quintosa					
<b>Descripción:</b> Se debe mostrar una tabla con el mes y el importe de salario mensual. Además, debe sumarse el importe de cada mes quedando el total del año.					
<b>Observaciones:</b> Ninguna					
<b>Prototipo:</b>					
<div style="text-align: center;"> <span>Registro de ingresos ▾</span> <span>Registro de gastos ▾</span> <span>Registro de bonificaciones ▾</span> <span>Registro de personal ▾</span> <span>Calculo de impuestos ▾</span> <span>Disponibilidad ▾</span> </div> <p style="text-align: center;">Listado de gastos de salario</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Mes</th> <th style="width: 50%;">Importe de salario</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Enero</td> <td style="text-align: center;">0,00</td> </tr> </tbody> </table>		Mes	Importe de salario	Enero	0,00
Mes	Importe de salario				
Enero	0,00				

Tabla A1.44 Historia de Usuario # 17.

Historia de Usuario									
<b>Número: 17</b>	<b>Nombre:</b> Listar los gastos totales								
<b>Usuario:</b> Usuario									
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto								
<b>Puntos estimados: 0.4</b>	<b>Iteración asignada: 3</b>								
<b>Programador responsable:</b> Dayanis Fundora Quintosa									
<b>Descripción:</b> Debe aparecer una tabla en la cual el usuario pueda ver los datos: mes, gastos por arrendamiento, gastos por espacios, total de gastos por arrendamiento y espacios con totales mensuales y anuales.									
<b>Observaciones:</b> En esta tabla el usuario solo debe ver el total mensual de gastos de ambas actividades y el anual. Los datos que aparecerían en esta tabla deben tomarse del registro de gastos de arrendamiento de habitaciones y del registro de gastos por el arrendamiento de espacios.									
<b>Prototipo:</b>									
<div style="text-align: center;"> <span>Registro de ingresos ▾</span> <span>Registro de gastos ▾</span> <span>Registro de bonificaciones ▾</span> <span>Registro de personal ▾</span> <span>Calculo de impuestos ▾</span> <span>Disponibilidad ▾</span> </div> <p style="text-align: center;">Listado de gastos totales de arrendamiento y espacio</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Mes</th> <th style="width: 25%;">Gasto por arrendamiento</th> <th style="width: 25%;">Gasto por espacio</th> <th style="width: 35%;">Total de gastos por arrendamiento y espacio</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Enero</td> <td style="text-align: center;">0,00</td> <td style="text-align: center;">0,00</td> <td style="text-align: center;">0,00</td> </tr> </tbody> </table>		Mes	Gasto por arrendamiento	Gasto por espacio	Total de gastos por arrendamiento y espacio	Enero	0,00	0,00	0,00
Mes	Gasto por arrendamiento	Gasto por espacio	Total de gastos por arrendamiento y espacio						
Enero	0,00	0,00	0,00						

Tabla A1.45 Historia de Usuario # 18.

Historia de Usuario	
<b>Número: 18</b>	<b>Nombre:</b> Cumplimentar libro de ingresos y gastos por el arrendamiento de espacios y de habitaciones
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto

<b>Puntos estimados: 1</b>	<b>Iteración asignada: 4</b>																																																												
<b>Programador responsable:</b> Dayanis Fundora Quintosa																																																													
<b>Descripción:</b> Se deben calcular automáticamente cada uno de los importes de ingresos y gastos diarios a partir de la información que introdujo el usuario en el “Registro de clientes” y “Registro de Gastos”																																																													
<b>Observaciones:</b> Ninguna																																																													
<b>Prototipo:</b>																																																													
<div style="text-align: center;"> <p>Ingresos de arrendamiento de habitaciones</p> <table border="1"> <thead> <tr> <th colspan="12">Ingresos</th> </tr> <tr> <th>D</th><th>Ene</th><th>D</th><th>Feb</th><th>D</th><th>Mar</th><th>D</th><th>Abr</th><th>D</th><th>May</th><th>D</th><th>Jun</th><th>D</th><th>Jul</th><th>D</th><th>Ago</th><th>D</th><th>Sep</th><th>D</th><th>Oct</th><th>D</th><th>Nov</th><th>D</th><th>Dic</th> </tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </tbody> </table> </div>		Ingresos												D	Ene	D	Feb	D	Mar	D	Abr	D	May	D	Jun	D	Jul	D	Ago	D	Sep	D	Oct	D	Nov	D	Dic																								
Ingresos																																																													
D	Ene	D	Feb	D	Mar	D	Abr	D	May	D	Jun	D	Jul	D	Ago	D	Sep	D	Oct	D	Nov	D	Dic																																						
<div style="text-align: center;"> <p>Gastos de arrendamiento de habitaciones</p> <table border="1"> <thead> <tr> <th colspan="12">Gastos</th> </tr> <tr> <th>D</th><th>Ene</th><th>D</th><th>Feb</th><th>D</th><th>Mar</th><th>D</th><th>Abr</th><th>D</th><th>May</th><th>D</th><th>Jun</th><th>D</th><th>Jul</th><th>D</th><th>Ago</th><th>D</th><th>Sep</th><th>D</th><th>Oct</th><th>D</th><th>Nov</th><th>D</th><th>Dic</th> </tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </tbody> </table> </div>		Gastos												D	Ene	D	Feb	D	Mar	D	Abr	D	May	D	Jun	D	Jul	D	Ago	D	Sep	D	Oct	D	Nov	D	Dic																								
Gastos																																																													
D	Ene	D	Feb	D	Mar	D	Abr	D	May	D	Jun	D	Jul	D	Ago	D	Sep	D	Oct	D	Nov	D	Dic																																						

Tabla A1.46 Historia de Usuario # 19.

<b>Historia de Usuario</b>																										
<b>Número: 19</b>	<b>Nombre:</b> Cumplimentar vector fiscal																									
<b>Usuario:</b> Usuario																										
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto																									
<b>Puntos estimados: 0.6</b>	<b>Iteración asignada: 4</b>																									
<b>Programador responsable:</b> Dayanis Fundora Quintosa																										
<b>Descripción:</b> Debe aparecer una tabla con los datos establecidos por la ONAT. En cada fila se debe cumplimentar el importe correspondiente según el código de tributo que tenga. Debe ser reflejada toda la información de cada tributo como lo establece la ONAT. La columna “Pagado” debe ser habilitada con <i>checkbox</i> permitiéndole al usuario marcar el importe que ya haya pagado.																										
<b>Observaciones:</b> Todos los importes de la tabla deben ser cumplimentados automáticamente a partir de los ingresos registrados por el usuario.																										
<b>Prototipo:</b>																										
<div style="text-align: center;"> <p>RC-04A Vector fiscal de persona natural</p> <p>NIT: 86110305809      Carné de identidad: 86110305809</p> <p>DPA-Municipio: 2304-Habana Vieja</p> <p>Código de barras del RC-05 (DPA, NIT): 0850901686014032</p> <p><b>Año: 2023</b></p> <table border="1"> <thead> <tr> <th colspan="5">Obligaciones tributarias en CUP(pesos cubanos)</th> </tr> <tr> <th>Pagado</th><th>Código barras</th><th>Importe</th><th>Código tributo</th><th>Período: Fecha límite de pago</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td><td>70013</td><td>0</td><td>0114022</td><td>Enero: 20/Feb/23</td> </tr> <tr> <td><input checked="" type="checkbox"/></td><td>00043</td><td>0</td><td>0510122</td><td>Enero: 20/Feb/23</td> </tr> <tr> <td><input checked="" type="checkbox"/></td><td>30303</td><td>0</td><td>0520522</td><td>Enero: 20/Feb/23</td> </tr> </tbody> </table> </div>		Obligaciones tributarias en CUP(pesos cubanos)					Pagado	Código barras	Importe	Código tributo	Período: Fecha límite de pago	<input checked="" type="checkbox"/>	70013	0	0114022	Enero: 20/Feb/23	<input checked="" type="checkbox"/>	00043	0	0510122	Enero: 20/Feb/23	<input checked="" type="checkbox"/>	30303	0	0520522	Enero: 20/Feb/23
Obligaciones tributarias en CUP(pesos cubanos)																										
Pagado	Código barras	Importe	Código tributo	Período: Fecha límite de pago																						
<input checked="" type="checkbox"/>	70013	0	0114022	Enero: 20/Feb/23																						
<input checked="" type="checkbox"/>	00043	0	0510122	Enero: 20/Feb/23																						
<input checked="" type="checkbox"/>	30303	0	0520522	Enero: 20/Feb/23																						

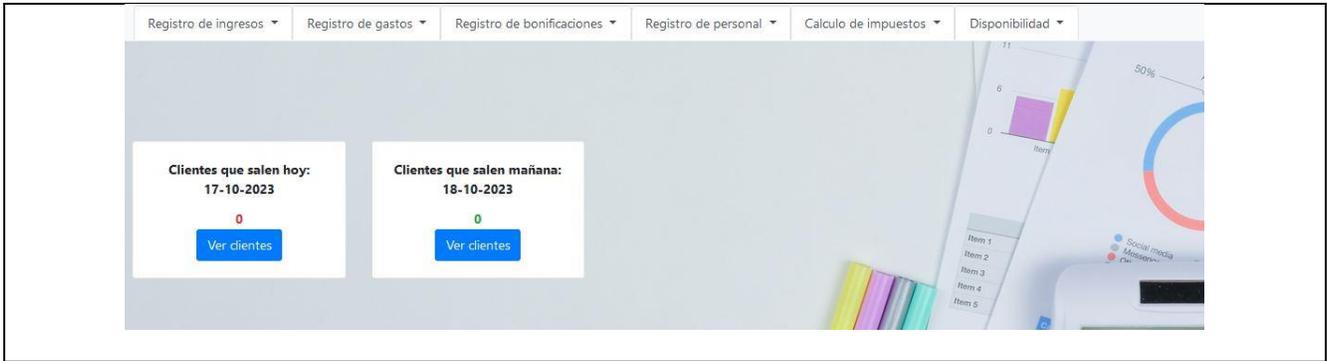
Tabla A1.47 Historia de Usuario # 20.

<b>Historia de Usuario</b>	
<b>Número: 20</b>	<b>Nombre:</b> Cumplimentar declaración jurada

<b>Usuario:</b> Usuario																																																																																																																																																																																																																		
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto																																																																																																																																																																																																																	
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 4																																																																																																																																																																																																																	
<b>Programador responsable:</b> Dayanis Fundora Quintosa																																																																																																																																																																																																																		
<b>Descripción:</b> Este modelo lo emite la ONAT y debe aparecer en el sistema cumplimentado automáticamente a partir de los registros de ingresos, gastos, propietario y trabajadores contratados que haya realizado el usuario.																																																																																																																																																																																																																		
<b>Observaciones:</b> Todos los datos de la tabla deben ser cumplimentados automáticamente.																																																																																																																																																																																																																		
<b>Prototipo:</b>																																																																																																																																																																																																																		
<div style="display: flex; justify-content: space-between; margin-bottom: 10px;"> <span>Registro de ingresos ▾</span> <span>Registro de gastos ▾</span> <span>Registro de bonificaciones ▾</span> <span>Registro de personal ▾</span> <span>Calculo de impuestos ▾</span> <span>Disponibilidad ▾</span> </div> <p style="text-align: center;">DJ-08 "Declaración Jurada"</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="3">ONAT</th> <th colspan="13">DECLARACIÓN JURADA IMPUESTO SOBRE INGRESOS PERSONALES - PESOS CUP</th> </tr> </thead> <tbody> <tr> <td>Año fiscal</td> <td>Individual</td> <td>X</td> <td>NIT</td> <td>9</td><td>7</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>8</td><td>2</td><td>5</td><td>5</td> <td>Rectificación DJ</td> </tr> <tr> <td></td> <td>Conjunta</td> <td>-</td> <td>Carné de identidad</td> <td>9</td><td>7</td><td>0</td><td>2</td><td>0</td><td>1</td><td>0</td><td>8</td><td>2</td><td>5</td><td>5</td> <td>X</td> </tr> <tr> <td colspan="3">Nombre (s) y apellidos:</td> <td colspan="13">Dayanis Fundora</td> </tr> <tr> <td colspan="3">Dirección según Carné de identidad</td> <td>Calle o Avenida</td> <td>Número</td> <td colspan="2">Apto</td> <td colspan="8">Entrecalles</td> </tr> <tr> <td colspan="3"></td> <td>Conde</td> <td>12</td> <td colspan="2">-</td> <td colspan="8">Bayona y Compostela</td> </tr> <tr> <td colspan="3">Referencia</td> <td>Reparto</td> <td colspan="4">Municipio</td> <td colspan="5">Provincia</td> </tr> <tr> <td colspan="3">-</td> <td>San Isidro</td> <td colspan="4">Habana Vieja</td> <td colspan="5">La Habana</td> </tr> <tr> <td>Código Zona Postal</td> <td>Teléfono</td> <td>Correo Electrónico</td> <td colspan="4">Opera en su municipio</td> <td colspan="5">Municipio donde opera</td> </tr> <tr> <td>10500</td> <td>53547753</td> <td>dayanis@gmail.com</td> <td>Sí</td> <td>-</td> <td>No</td> <td>-</td> <td colspan="5">Habana Vieja</td> </tr> <tr> <td colspan="4">Si es usted del Régimen Simplificado de tributación marque:</td> <td colspan="2">Sí:</td> <td>X</td> <td colspan="2">NO</td> <td colspan="4">-</td> </tr> <tr> <td colspan="3">Impuesto sobre Ingresos Personales. Liquidación adicional.</td> <td colspan="5">Código del tributo</td> <td colspan="5">Código del Banco</td> </tr> <tr> <td colspan="3"></td> <td colspan="5">053022-2</td> <td colspan="5">30102</td> </tr> <tr> <td colspan="3"></td> <td colspan="5">TOTAL A PAGAR</td> <td colspan="5">Importe fila 36</td> </tr> <tr> <td colspan="3"></td> <td colspan="5"></td> <td colspan="5">0</td> </tr> </tbody> </table> <p><b>OBJETIVO:</b> Formalizar mediante Declaración Jurada la liquidación y pago en Pesos - CUP, anual o de un período menor de 12 meses al causar baja el contribuyente, del impuesto sobre Ingresos Personales, que deben presentar todas las personas</p>		ONAT			DECLARACIÓN JURADA IMPUESTO SOBRE INGRESOS PERSONALES - PESOS CUP													Año fiscal	Individual	X	NIT	9	7	0	2	0	1	0	8	2	5	5	Rectificación DJ		Conjunta	-	Carné de identidad	9	7	0	2	0	1	0	8	2	5	5	X	Nombre (s) y apellidos:			Dayanis Fundora													Dirección según Carné de identidad			Calle o Avenida	Número	Apto		Entrecalles											Conde	12	-		Bayona y Compostela								Referencia			Reparto	Municipio				Provincia					-			San Isidro	Habana Vieja				La Habana					Código Zona Postal	Teléfono	Correo Electrónico	Opera en su municipio				Municipio donde opera					10500	53547753	dayanis@gmail.com	Sí	-	No	-	Habana Vieja					Si es usted del Régimen Simplificado de tributación marque:				Sí:		X	NO		-				Impuesto sobre Ingresos Personales. Liquidación adicional.			Código del tributo					Código del Banco								053022-2					30102								TOTAL A PAGAR					Importe fila 36													0				
ONAT			DECLARACIÓN JURADA IMPUESTO SOBRE INGRESOS PERSONALES - PESOS CUP																																																																																																																																																																																																															
Año fiscal	Individual	X	NIT	9	7	0	2	0	1	0	8	2	5	5	Rectificación DJ																																																																																																																																																																																																			
	Conjunta	-	Carné de identidad	9	7	0	2	0	1	0	8	2	5	5	X																																																																																																																																																																																																			
Nombre (s) y apellidos:			Dayanis Fundora																																																																																																																																																																																																															
Dirección según Carné de identidad			Calle o Avenida	Número	Apto		Entrecalles																																																																																																																																																																																																											
			Conde	12	-		Bayona y Compostela																																																																																																																																																																																																											
Referencia			Reparto	Municipio				Provincia																																																																																																																																																																																																										
-			San Isidro	Habana Vieja				La Habana																																																																																																																																																																																																										
Código Zona Postal	Teléfono	Correo Electrónico	Opera en su municipio				Municipio donde opera																																																																																																																																																																																																											
10500	53547753	dayanis@gmail.com	Sí	-	No	-	Habana Vieja																																																																																																																																																																																																											
Si es usted del Régimen Simplificado de tributación marque:				Sí:		X	NO		-																																																																																																																																																																																																									
Impuesto sobre Ingresos Personales. Liquidación adicional.			Código del tributo					Código del Banco																																																																																																																																																																																																										
			053022-2					30102																																																																																																																																																																																																										
			TOTAL A PAGAR					Importe fila 36																																																																																																																																																																																																										
								0																																																																																																																																																																																																										

Tabla A1.48 Historia de Usuario # 21.

<b>Historia de Usuario</b>	
<b>Número:</b> 21	<b>Nombre:</b> Mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día
<b>Usuario:</b> Usuario	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 0.1	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Dayanis Fundora Quintosa	
<b>Descripción:</b> Mostrar al usuario en la pantalla de inicio la cantidad de clientes que salen en la fecha actual y al día siguiente, con un botón que le permita ver el listado con los nombres, fecha de entrada, fecha de salida y cantidad de noches de dichos clientes.	
<b>Observaciones:</b> Ninguna	
<b>Prototipo:</b>	



## A.2. Iteración 1

## HU3: Gestionar cliente

Tabla A2.49 Tarea de ingeniería #6.

Tarea	
<b>Número de la tarea:</b> 6	<b>Número de Historia de usuario:</b> 3
<b>Nombre de la tarea:</b> Listar clientes	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 5 de junio de 2023	<b>Fecha de fin:</b> 5 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para listar clientes, manejar mensajes en caso de que no existan clientes registrados.	

Tabla A2.50 Tarea de ingeniería #7.

Tarea	
<b>Número de la tarea:</b> 7	<b>Número de Historia de usuario:</b> 3
<b>Nombre de la tarea:</b> Adicionar clientes	<b>Puntos de estimación:</b> 0.6
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 6 de junio de 2023	<b>Fecha de fin:</b> 9 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista, formulario y url para adicionar un nuevo cliente, mostrar los campos del formulario de registro, manejar mensajes en caso de registro satisfactorio o de formulario no válido, redireccionar a la url registrar clientes en caso satisfactorio.	

Tabla A2.51 Tarea de ingeniería #8.

Tarea	
<b>Número de la tarea:</b> 8	<b>Número de Historia de usuario:</b> 3
<b>Nombre de la tarea:</b> Editar cliente	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 10 de junio de 2023	<b>Fecha de fin:</b> 11 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista, formulario y url para editar un cliente registrado, mostrar los campos del formulario de actualización, redireccionar a la url listar clientes en caso satisfactorio.	

Tabla A2.52 Tarea de ingeniería #9.

Tarea	
<b>Número de la tarea:</b> 9	<b>Número de Historia de usuario:</b> 3
<b>Nombre de la tarea:</b> Eliminar cliente	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 12 de junio de 2023	<b>Fecha de fin:</b> 12 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar un cliente registrado, mostrar confirmación en forma de modal, redireccionar a la url listado de clientes registrados en caso satisfactorio.	

## HU4: Gestionar el ingreso por arrendamiento de espacios

Tabla A2.53 Tarea de ingeniería #10.

Tarea	
<b>Número de la tarea:</b> 10	<b>Número de Historia de usuario:</b> 4
<b>Nombre de la tarea:</b> Listar espacios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 13 de junio de 2023	<b>Fecha de fin:</b> 13 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para listar espacios, manejar mensajes en caso de que no existan espacios registrados.	

Tabla A2.54 Tarea de ingeniería #11.

Tarea	
<b>Número de la tarea:</b> 11	<b>Número de Historia de usuario:</b> 4
<b>Nombre de la tarea:</b> Adicionar espacios	<b>Puntos de estimación:</b> 0.4
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 14 de junio de 2023	<b>Fecha de fin:</b> 16 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista, formulario y url para adicionar un nuevo espacio, mostrar los campos del formulario de registro en forma escalonada, manejar mensajes en caso de registro satisfactorio o de formulario no válido, redireccionar a la url registro de espacios en caso satisfactorio.	

Tabla A2.55 Tarea de ingeniería #12.

Tarea	
<b>Número de la tarea:</b> 12	<b>Número de Historia de usuario:</b> 4
<b>Nombre de la tarea:</b> Editar espacios	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 17 de junio de 2023	<b>Fecha de fin:</b> 18 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista, formulario y url para editar un espacio registrado, mostrar los campos del formulario de actualización en forma escalonada, redireccionar a la url listado de ingresos de espacios en caso satisfactorio.	

Tabla A2.56 Tarea de ingeniería #13.

Tarea	
<b>Número de la tarea:</b> 13	<b>Número de Historia de usuario:</b> 4
<b>Nombre de la tarea:</b> Eliminar espacios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 19 de junio de 2023	<b>Fecha de fin:</b> 19 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar un espacio registrado, mostrar confirmación en forma de modal, redireccionar a la url listado de ingresos de espacios en caso satisfactorio.	

## HU5: Gestionar las bonificaciones por pago electrónico

Tabla A2.57 Tarea de Ingeniería # 14.

Tarea	
<b>Número de la tarea:</b> 14	<b>Número de Historia de usuario:</b> 5
<b>Nombre de la tarea:</b> Listar bonificaciones por pago electrónico	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	

<b>Fecha de inicio:</b> 15 de agosto de 2023	<b>Fecha de fin:</b> 15 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla en la que el usuario pueda ver todas las bonificaciones por pago electrónico.	

Tabla A2.58 Tarea de Ingeniería #15.

Tarea	
<b>Número de la tarea:</b> 15	<b>Número de Historia de usuario:</b> 5
<b>Nombre de la tarea:</b> Adicionar bonificaciones por pago electrónico	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 16 de agosto de 2023	<b>Fecha de fin:</b> 17 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear formulario donde el usuario adicione todas las bonificaciones por pago electrónico.	

Tabla A2.59 Tarea de Ingeniería #16.

Tarea	
<b>Número de la tarea:</b> 16	<b>Número de Historia de usuario:</b> 5
<b>Nombre de la tarea:</b> Editar bonificaciones por pago electrónico	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 18 de agosto de 2023	<b>Fecha de fin:</b> 18 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear la opción para que el usuario pueda modificar las bonificaciones por pago electrónico.	

Tabla A2.60 Tarea de Ingeniería #17.

Tarea	
<b>Número de la tarea:</b> 17	<b>Número de Historia de usuario:</b> 5
<b>Nombre de la tarea:</b> Eliminar bonificaciones por pago electrónico	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 19 de agosto de 2023	<b>Fecha de fin:</b> 19 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar las bonificaciones por pago electrónico.	

### A.3. Iteración 2

#### HU6: Gestionar gastos por arrendamiento de espacios

Tabla A3.61 Tarea de Ingeniería #18.

Tarea	
<b>Número de la tarea:</b> 18	<b>Número de Historia de usuario:</b> 6
<b>Nombre de la tarea:</b> Listar gastos de arrendamiento de espacios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 04 de julio de 2023	<b>Fecha de fin:</b> 04 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla en la que el usuario pueda ver todos los gastos por el arrendamiento de espacios registrados.	

Tabla A3.62 Tarea de Ingeniería #19.

Tarea	
<b>Número de la tarea:</b> 19	<b>Número de Historia de usuario:</b> 6
<b>Nombre de la tarea:</b> Adicionar gastos por el arrendamiento de espacios	<b>Puntos de estimación:</b> 0.6
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 05 de julio de 2023	<b>Fecha de fin:</b> 08 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear formulario donde el usuario adicione los gastos por arrendamiento de espacios que tiene.	

Tabla A3.63 Tarea de Ingeniería #20.

Tarea	
<b>Número de la tarea:</b> 20	<b>Número de Historia de usuario:</b> 6
<b>Nombre de la tarea:</b> Editar gastos por el arrendamiento de espacios	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 09 de julio de 2023	<b>Fecha de fin:</b> 10 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear la opción para que el usuario pueda modificar el gasto por el arrendamiento de espacios que elija.	

Tabla A3.64 Tarea de Ingeniería #21.

Tarea	
<b>Número de la tarea:</b> 21	<b>Número de Historia de usuario:</b> 6
<b>Nombre de la tarea:</b> Eliminar gasto por el arrendamiento de espacios	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 11 de julio de 2023	<b>Fecha de fin:</b> 11 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar un gasto registrado, mostrar confirmación en forma de modal, redireccionar a la url listado de gastos por el arrendamiento de espacios.	

#### HU7: Gestionar gastos por arrendamiento de habitaciones

Tabla A3.65 Tarea de Ingeniería #22.

Tarea	
<b>Número de la tarea:</b> 22	<b>Número de Historia de usuario:</b> 7
<b>Nombre de la tarea:</b> Listar gastos por el arrendamiento de habitaciones	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 26 de junio de 2023	<b>Fecha de fin:</b> 26 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla en la que el usuario pueda ver todos los gastos por el arrendamiento de habitaciones registrados.	

Tabla A3.66 Tarea de Ingeniería #23.

Tarea	
<b>Número de la tarea:</b> 23	<b>Número de Historia de usuario:</b> 7
<b>Nombre de la tarea:</b> Adicionar gastos por el arrendamiento de habitaciones	<b>Puntos de estimación:</b> 0.6
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 27 de junio de 2023	<b>Fecha de fin:</b> 30 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear formulario donde el usuario adicione los gastos por arrendamiento de habitaciones que tiene.	

Tabla A3.67 Tarea de Ingeniería #24.

Tarea	
<b>Número de la tarea:</b> 24	<b>Número de Historia de usuario:</b> 7
<b>Nombre de la tarea:</b> Editar gastos por el arrendamiento de habitaciones	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 01 de julio de 2023	<b>Fecha de fin:</b> 02 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear la opción para que el usuario pueda modificar el gasto que elija.	

Tabla A3.68 Tarea de Ingeniería #25.

Tarea	
<b>Número de la tarea:</b> 25	<b>Número de Historia de usuario:</b> 7
<b>Nombre de la tarea:</b> Eliminar gasto por el arrendamiento de habitaciones	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 03 de julio de 2023	<b>Fecha de fin:</b> 03 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar un gasto registrado, mostrar confirmación en forma de modal, redireccionar a la url listado de gastos por el arrendamiento de habitaciones.	

## HU8: Gestionar propietario

Tabla A3.69 Tarea de Ingeniería #26.

Tarea	
<b>Número de la tarea:</b> 26	<b>Número de Historia de usuario:</b> 8
<b>Nombre de la tarea:</b> Listar propietario	<b>Puntos de estimación:</b> 0.1

<b>Tipo de tarea: Desarrollo</b>	
<b>Fecha de inicio:</b> 20 de agosto de 2023	<b>Fecha de fin:</b> 20 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla en la que el usuario pueda ver los datos del propietario.	

Tabla A3.70 Tarea de Ingeniería #27.

<b>Tarea</b>	
<b>Número de la tarea:</b> 27	<b>Número de Historia de usuario:</b> 8
<b>Nombre de la tarea:</b> Adicionar propietario	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea: Desarrollo</b>	
<b>Fecha de inicio:</b> 21 de agosto de 2023	<b>Fecha de fin:</b> 22 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear formulario donde el usuario adicione los datos del propietario.	

Tabla A3.71 Tarea de Ingeniería #28.

<b>Tarea</b>	
<b>Número de la tarea:</b> 28	<b>Número de Historia de usuario:</b> 8
<b>Nombre de la tarea:</b> Editar propietario	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea: Desarrollo</b>	
<b>Fecha de inicio:</b> 23 de agosto de 2023	<b>Fecha de fin:</b> 23 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear la opción para que el usuario pueda modificar los datos del propietario.	

Tabla A3.72 Tarea de Ingeniería #29.

<b>Tarea</b>	
<b>Número de la tarea:</b> 29	<b>Número de Historia de usuario:</b> 8
<b>Nombre de la tarea:</b> Eliminar propietario	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea: Desarrollo</b>	
<b>Fecha de inicio:</b> 24 de agosto de 2023	<b>Fecha de fin:</b> 24 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar los datos del propietario.	

## HU9: Gestionar trabajadores contratados

Tabla A3.73 Tarea de Ingeniería #30.

<b>Tarea</b>	
<b>Número de la tarea:</b> 30	<b>Número de Historia de usuario:</b> 9
<b>Nombre de la tarea:</b> Listar trabajadores contratados	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea: Desarrollo</b>	
<b>Fecha de inicio:</b> 25 de agosto de 2023	<b>Fecha de fin:</b> 25 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla en la que el usuario pueda ver los datos de los trabajadores contratados.	

Tabla A3.74 Tarea de Ingeniería #31.

<b>Tarea</b>	
<b>Número de la tarea:</b> 31	<b>Número de Historia de usuario:</b> 9
<b>Nombre de la tarea:</b> Adicionar trabajadores contratados	<b>Puntos de estimación:</b> 0.3

<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 26 de agosto de 2023	<b>Fecha de fin:</b> 27 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear formulario donde el usuario adicione los datos de los trabajadores contratados.	

Tabla A3.75 Tarea de Ingeniería #32.

<b>Tarea</b>	
<b>Número de la tarea:</b> 32	<b>Número de Historia de usuario:</b> 9
<b>Nombre de la tarea:</b> Editar trabajadores contratados	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 28 de agosto de 2023	<b>Fecha de fin:</b> 28 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear la opción para que el usuario pueda modificar los datos de los trabajadores contratados	

Tabla A3.76 Tarea de Ingeniería #33.

<b>Tarea</b>	
<b>Número de la tarea:</b> 33	<b>Número de Historia de usuario:</b> 9
<b>Nombre de la tarea:</b> Eliminar trabajadores contratados	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 29 de agosto de 2023	<b>Fecha de fin:</b> 29 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para eliminar los datos de los trabajadores contratados.	

## HU10: Calcular el índice ocupacional

Tabla A3.77 Tarea de Ingeniería #34.

<b>Tarea</b>	
<b>Número de la tarea:</b> 34	<b>Número de Historia de usuario:</b> 10
<b>Nombre de la tarea:</b> Desarrollar el cálculo del índice ocupacional	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos de estimación:</b> 0.4
<b>Fecha de inicio:</b> 20 de junio de 2023	<b>Fecha de fin:</b> 22 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear tabla en la que se calcule automáticamente el índice ocupacional mensual a partir de la cantidad de clientes que se hayan registrado y de la capacidad de arrendamiento existente.	

## A.4. Iteración 3

## HU11: Calcular la disponibilidad de arrendamiento por habitaciones

Tabla A4.78 Tarea de Ingeniería #35.

Tarea	
<b>Número de la tarea:</b> 35	<b>Número de Historia de usuario:</b> 11
<b>Nombre de la tarea:</b> Desarrollar el cálculo de la disponibilidad de arrendamiento por habitaciones	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 18 de julio de 2023	<b>Fecha de fin:</b> 19 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre la cantidad de clientes hospedados, la capacidad por habitaciones y la disponibilidad existente.	

## HU12: Calcular la disponibilidad de arrendamiento por camas

Tabla A4.79 Tarea de Ingeniería #36.

Tarea	
<b>Número de la tarea:</b> 36	<b>Número de Historia de usuario:</b> 12
<b>Nombre de la tarea:</b> Desarrollar el cálculo de la disponibilidad de arrendamiento por camas	<b>Puntos de estimación:</b> 0.3
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 20 de julio de 2023	<b>Fecha de fin:</b> 21 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre la cantidad de clientes hospedados, la capacidad por camas y la disponibilidad existente.	

## HU13: Calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones

Tabla A4.80 Tarea de Ingeniería #37.

Tarea	
<b>Número de la tarea:</b> 37	<b>Número de Historia de usuario:</b> 13
<b>Nombre de la tarea:</b> Desarrollar el cálculo de los importes correspondientes al pago de tributo por el arrendamiento de habitaciones	<b>Puntos de estimación:</b> 1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 29 de julio de 2023	<b>Fecha de fin:</b> 04 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre una tabla con los importes a pagar a la ONAT según lo establecido.	

## HU14: Calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios

Tabla A4.81 Tarea de Ingeniería #38.

Tarea	
<b>Número de la tarea:</b> 38	<b>Número de Historia de usuario:</b> 14
<b>Nombre de la tarea:</b> Desarrollar el cálculo de los importes correspondientes al pago de tributo por el arrendamiento de espacios	<b>Puntos de estimación:</b> 1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 04 de agosto de 2023	<b>Fecha de fin:</b> 10 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre una tabla con los importes a pagar a la ONAT según lo establecido.	

### HU15: Listar los ingresos totales

Tabla A4.82 Tarea de Ingeniería #39.

Tarea	
<b>Número de la tarea:</b> 39	<b>Número de Historia de usuario:</b> 15
<b>Nombre de la tarea:</b> Desarrollar los ingresos totales	<b>Puntos de estimación:</b> 0.4
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 23 de junio de 2023	<b>Fecha de fin:</b> 25 de junio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear tabla donde se calcule automáticamente el ingreso total mensual y anual por arrendamiento de habitaciones, de espacios y el total de ambos.	

### HU16: Listar gastos de salario

Tabla A4.83 Tarea de Ingeniería #40.

Tarea	
<b>Número de la tarea:</b> 40	<b>Número de Historia de usuario:</b> 16
<b>Nombre de la tarea:</b> Desarrollar gasto de salario	<b>Puntos de estimación:</b> 0.4
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 12 de julio de 2023	<b>Fecha de fin:</b> 14 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla donde aparezca el total de gastos mensual por concepto de salario que haya sido registrado.	

### HU17: Listar los gastos totales

Tabla A4.84 Tarea de Ingeniería #41.

Tarea	
<b>Número de la tarea:</b> 41	<b>Número de Historia de usuario:</b> 17
<b>Nombre de la tarea:</b> Desarrollar gastos totales	<b>Puntos de estimación:</b> 0.4
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 15 de julio de 2023	<b>Fecha de fin:</b> 17 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear una tabla donde aparezcan los gastos totales mensuales y anuales de la actividad de arrendamiento de habitaciones y de espacios.	

## A.5. Iteración 4

## HU18: Cumplimentar libro de ingresos y gastos por el arrendamiento de espacios y de habitaciones

Tabla A5.85 Tarea de Ingeniería #42.

Tarea	
<b>Número de la tarea:</b> 42	<b>Número de Historia de usuario:</b> 18
<b>Nombre de la tarea:</b> Desarrollar libro de ingresos y gastos	<b>Puntos de estimación:</b> 1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 22 de julio de 2023	<b>Fecha de fin:</b> 28 de julio de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre una tabla con los importes diarios de ingresos y gastos registrados por el usuario.	

## HU19: Cumplimentar vector fiscal

Tabla A5.86 Tarea de Ingeniería #43.

Tarea	
<b>Número de la tarea:</b> 43	<b>Número de Historia de usuario:</b> 19
<b>Nombre de la tarea:</b> Desarrollar vector fiscal	<b>Puntos de estimación:</b> 0.6
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 11 de agosto de 2023	<b>Fecha de fin:</b> 14 de agosto de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre una tabla con los importes a pagar a la ONAT según lo establecido para el arrendamiento de espacios y de habitaciones desglosado por código de tributo.	

## HU20: Cumplimentar declaración jurada

Tabla A5.87 Tarea de Ingeniería #44.

Tarea	
<b>Número de la tarea:</b> 44	<b>Número de Historia de usuario:</b> 20
<b>Nombre de la tarea:</b> Desarrollar declaración jurada	<b>Puntos de estimación:</b> 1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 30 de agosto de 2023	<b>Fecha de fin:</b> 05 de septiembre de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre una tabla dividida en secciones, con los importes, que establece el modelo, cumplimentados automáticamente, correspondientes al pago anual de las obligaciones fiscales.	

## HU21: Mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día

Tabla A5.88 Tarea de Ingeniería #45.

Tarea	
<b>Número de la tarea:</b> 45	<b>Número de Historia de usuario:</b> 21

<b>Nombre de la tarea:</b> Desarrollar alerta de finalización de estancia	<b>Puntos de estimación:</b> 0.1
<b>Tipo de tarea:</b> Desarrollo	
<b>Fecha de inicio:</b> 06 de septiembre de 2023	<b>Fecha de fin:</b> 06 de septiembre de 2023
<b>Programador responsable:</b> Dayanis Fundora	
<b>Descripción:</b> Crear plantilla, vista y url para que se muestre la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día. Además, mostrar la lista de dichos clientes cuando el usuario presione la opción "ver clientes".	

## A.6. Pruebas de aceptación - Iteración 1

Tabla A6.89 Prueba de aceptación #3.

Caso de prueba de aceptación	
<b>Código:</b> HU3_P3	<b>Historia de Usuario:</b> 3
<b>Nombre:</b> Gestionar cliente	
<b>Descripción:</b> Prueba para la funcionalidad gestionar cliente.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un cliente en el sistema con los datos válidos. Se intenta registrar un cliente en el sistema con los datos inválidos. Se intenta modificar un cliente registrado en el sistema con datos válidos. Se intenta modificar un cliente registrado en el sistema con datos inválidos. Se intenta mostrar un cliente registrado. Se intenta eliminar un cliente del sistema.	
<b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.	

Tabla A6.90 Prueba de aceptación #4.

Caso de prueba de aceptación	
<b>Código:</b> HU4_P4	<b>Historia de Usuario:</b> 4
<b>Nombre:</b> Gestionar el ingreso por arrendamiento de espacios	
<b>Descripción:</b> Prueba para la funcionalidad gestionar el ingreso por arrendamiento de espacios.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un ingreso de espacios en el sistema con los datos válidos. Se intenta registrar un ingreso de espacios en el sistema con los datos inválidos. Se intenta modificar un ingreso de espacios registrado en el sistema con datos válidos. Se intenta modificar un ingreso de espacios registrado en el sistema con datos inválidos. Se intenta mostrar un ingreso de espacios registrado. Se intenta eliminar un ingreso de espacios del sistema.	
<b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.	

Tabla A6.91 Prueba de aceptación #5.

Caso de prueba de aceptación	
<b>Código:</b> HU5_P5	<b>Historia de Usuario:</b> 5
<b>Nombre:</b> Gestionar las bonificaciones por pago electrónico	
<b>Descripción:</b> Prueba para la funcionalidad gestionar las bonificaciones por pago electrónico.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar una bonificación por pago electrónico en el sistema con los datos válidos. Se intenta registrar una bonificación por pago electrónico en el sistema con los datos inválidos. Se intenta modificar una bonificación por pago electrónico en el sistema con datos válidos. Se intenta modificar una bonificación por pago electrónico en el sistema con datos inválidos. Se intenta eliminar una bonificación por pago electrónico del sistema.	

**Resultados esperados:** Si los datos son correctos, se registra, se modifica y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.

## A.7. Pruebas de aceptación - Iteración 2

Tabla A7.92 Prueba de aceptación #6.

Caso de prueba de aceptación	
<b>Código:</b> HU6_P6	<b>Historia de Usuario:</b> 6
<b>Nombre:</b> Gestionar los gastos por el arrendamiento de espacios	
<b>Descripción:</b> Prueba para la funcionalidad gestionar los gastos por el arrendamiento de espacios.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un gasto por el arrendamiento de espacios en el sistema con los datos válidos. Se intenta registrar un gasto por el arrendamiento de espacios en el sistema con los datos inválidos. Se intenta modificar un gasto por el arrendamiento de espacios registrado en el sistema con datos válidos. Se intenta modificar un gasto por el arrendamiento de espacios registrado en el sistema con datos inválidos. Se intenta mostrar gasto por el arrendamiento de espacios registrado. Se intenta eliminar gasto por el arrendamiento de espacios del sistema.	
<b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.	

Tabla A7.93 Prueba de aceptación #7.

Caso de prueba de aceptación	
<b>Código:</b> HU7_P7	<b>Historia de Usuario:</b> 7
<b>Nombre:</b> Gestionar gastos por arrendamiento de habitaciones	
<b>Descripción:</b> Prueba para la funcionalidad gestionar gastos por arrendamiento de habitaciones.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un gasto por el arrendamiento de habitaciones en el sistema con los datos válidos. Se intenta registrar un gasto por el arrendamiento de habitaciones en el sistema con los datos inválidos. Se intenta modificar un gasto por el arrendamiento de habitaciones registrado en el sistema con datos válidos. Se intenta modificar un gasto por el arrendamiento de habitaciones registrado en el sistema con datos inválidos. Se intenta mostrar gasto por el arrendamiento de habitaciones registrado. Se intenta eliminar gasto por el arrendamiento de habitaciones del sistema.	
<b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.	

Tabla A7.94 Prueba de aceptación #8.

Caso de prueba de aceptación	
<b>Código:</b> HU8_P8	<b>Historia de Usuario:</b> 8
<b>Nombre:</b> Gestionar propietario	
<b>Descripción:</b> Prueba para la funcionalidad gestionar propietario.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un propietario en el sistema con los datos válidos. Se intenta registrar un propietario en el sistema con los datos inválidos.	

<p>Se intenta modificar un propietario registrado en el sistema con datos válidos.          Se intenta modificar un propietario registrado en el sistema con datos inválidos.          Se intenta mostrar un propietario registrado.          Se intenta eliminar un propietario del sistema.</p>
<p><b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.</p>

Tabla A7.95 Prueba de aceptación #9.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU9_P9	<b>Historia de Usuario:</b> 9
<b>Nombre:</b> Gestionar trabajadores contratados	
<b>Descripción:</b> Prueba para la funcionalidad gestionar cliente.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se intenta registrar un trabajador contratado en el sistema con los datos válidos. Se intenta registrar un trabajador contratado en el sistema con los datos inválidos. Se intenta modificar un trabajador contratado registrado en el sistema con datos válidos. Se intenta modificar un trabajador contratado registrado en el sistema con datos inválidos. Se intenta mostrar un trabajador contratado registrado. Se intenta eliminar un trabajador contratado del sistema.	
<b>Resultados esperados:</b> Si los datos son correctos, se registra, se modifica, se muestra y se elimina en cada caso. Si los datos son incorrectos se redirecciona para la misma página mostrando los errores.	

Tabla A7.96 Prueba de aceptación #10.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU10_P10	<b>Historia de Usuario:</b> 10
<b>Nombre:</b> Calcular índice ocupacional	
<b>Descripción:</b> Prueba para la funcionalidad calcular índice ocupacional.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los porcentajes mostrados.	
<b>Resultados esperados:</b> Que los porcentajes calculados sean los correctos.	

## A.8. Pruebas de aceptación - Iteración 3

Tabla A8.97 Prueba de aceptación #11.

Caso de prueba de aceptación	
<b>Código:</b> HU11_P11	<b>Historia de Usuario:</b> 11
<b>Nombre:</b> Calcular la disponibilidad de arrendamiento por habitaciones	
<b>Descripción:</b> Prueba para la funcionalidad calcular la disponibilidad de arrendamiento por habitaciones.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los datos mostrados.	
<b>Resultados esperados:</b> Que los datos calculados sean los correctos.	

Tabla A8.98 Prueba de aceptación #12.

Caso de prueba de aceptación	
<b>Código:</b> HU12_P12	<b>Historia de Usuario:</b> 12
<b>Nombre:</b> Calcular la disponibilidad de arrendamiento por camas	
<b>Descripción:</b> Prueba para la funcionalidad calcular la disponibilidad de arrendamiento por camas.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los datos mostrados.	
<b>Resultados esperados:</b> Que los datos calculados sean los correctos.	

Tabla A8.99 Prueba de aceptación #13.

Caso de prueba de aceptación	
<b>Código:</b> HU13_P13	<b>Historia de Usuario:</b> 13
<b>Nombre:</b> Calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones	
<b>Descripción:</b> Prueba para la funcionalidad calcular los importes correspondientes al pago de tributo por el arrendamiento de habitaciones	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean los correctos.	

Tabla A8.100 Prueba de aceptación #14.

Caso de prueba de aceptación	
<b>Código:</b> HU14_P14	<b>Historia de Usuario:</b> 14
<b>Nombre:</b> Calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios	
<b>Descripción:</b> Prueba para la funcionalidad calcular los importes correspondientes al pago de tributo por el arrendamiento de espacios.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean los correctos.	

Tabla A8.101 Prueba de aceptación #15.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU15_P15	<b>Historia de Usuario:</b> 15
<b>Nombre:</b> Listar los ingresos totales	
<b>Descripción:</b> Prueba para la funcionalidad listar los ingresos totales.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean correctos.	

Tabla A8.102 Prueba de aceptación #16.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU16_P16	<b>Historia de Usuario:</b> 16
<b>Nombre:</b> Listar gastos de salario	
<b>Descripción:</b> Prueba para la funcionalidad listar gastos de salario.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean correctos.	

Tabla A8.103 Prueba de aceptación #17.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU17_P17	<b>Historia de Usuario:</b> 17
<b>Nombre:</b> Listar los gastos totales	
<b>Descripción:</b> Prueba para la funcionalidad listar los gastos totales.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean correctos.	

## A.9. Pruebas de aceptación - Iteración 4

Tabla A9.104 Prueba de aceptación #18.

Caso de prueba de aceptación	
<b>Código:</b> HU18_P18	<b>Historia de Usuario:</b> 18
<b>Nombre:</b> Cumplimentar libro de ingresos y gastos por el arrendamiento de espacios y de habitaciones	
<b>Descripción:</b> Prueba para la funcionalidad cumplimentar libro de ingresos y gastos por el arrendamiento de espacios y de habitaciones.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean los correctos.	

Tabla A9.105 Prueba de aceptación #19.

Caso de prueba de aceptación	
<b>Código:</b> HU19_P19	<b>Historia de Usuario:</b> 19
<b>Nombre:</b> Cumplimentar vector fiscal	
<b>Descripción:</b> Prueba para la funcionalidad cumplimentar vector fiscal.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean los correctos.	

Tabla A9.106 Prueba de aceptación #20.

Caso de prueba de aceptación	
<b>Código:</b> HU20_P20	<b>Historia de Usuario:</b> 20
<b>Nombre:</b> Cumplimentar declaración jurada	
<b>Descripción:</b> Prueba para la funcionalidad cumplimentar la declaración jurada.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los importes mostrados.	
<b>Resultados esperados:</b> Que los importes calculados sean los correctos.	

Tabla A9.107 Prueba de aceptación #21

Caso de prueba de aceptación	
<b>Código:</b> HU21_P21	<b>Historia de Usuario:</b> 21
<b>Nombre:</b> Mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día	
<b>Descripción:</b> Prueba para la funcionalidad mostrar la cantidad de clientes que finalizan su estancia en la fecha actual y al siguiente día.	
<b>Condiciones de ejecución:</b> El usuario debe estar previamente autenticado.	
<b>Pasos de ejecución:</b> Se comprueba si son correctos los datos mostrados.	
<b>Resultados esperados:</b> Que los datos calculados sean los correctos.	

A.10. Entrevistas

No. Página:

## ENTREVISTA

Fecha:

Nombre: Leonel  
 Apellidos: Talledo Hernández

Actividad que realiza como TCP: Arrendador de Habitaciones

1. ¿Registra usted mismo a sus clientes en el Libro de Arredamiento?  SI  NO

1.1 En caso de que la respuesta sea NO especifique quién los registra?

Tenedor de libros  Familiar  Trabajador contratado

2. ¿Tiene conocimiento contable?  SI  NO

3. ¿Cumplimenta usted mismo su Declaración Jurada DJ-08 anualmente?  SI  NO

3.1 En caso de responder NO, especifique quién la cumple?

Tenedor de libros  Familiar  Trabajador contratado

4. ¿Confecciona usted la contabilidad de su negocio?  SI  NO

4.1 En caso de que la respuesta sea NO, especifique quién la confecciona:

Tenedor de libros  Familiar  Trabajador contratado

5. ¿Utiliza un sistema contable?  SI  NO

5.1 En caso de que NO utilice un sistema contable responda lo siguiente:

5.1.1 ¿Qué lo limita a usar un sistema contable?

Es muy caro  Desconocimiento de cuáles existen  Otros

En caso de marcar la opción "Otros" especifique cuál es la otra limitante:

El sistema que me han propuesto utilizar es monitoreado a distancia por personas desconocidas mi negocio, lo cual no me ha resultado confiable.

Figura A10.12 Entrevista #1

## ENTREVISTA

**¿Qué aspecto negativo considera que trae consigo la(s) limitante(s)?**

El aspecto negativo que traen consigo las limitantes expuestas anteriormente lo veo en que mi trabajador contratado debe realizar todos los cálculos a mano lo cual conlleva a invertir más tiempo, aumenta las posibilidades de errores y de pago indebido al fisco.

**6. ¿Le sería útil un sistema informático que le permita, a partir del registro de los clientes en un Libro de Arrendamiento digital, obtener el resultado del cálculo de los impuestos a pagar al fisco? ¿Por qué? ¿Considera que le traería algún beneficio?**

Un sistema informático que permita eso considero que sería muy útil sobre todo para los TCP que no tenemos conocimientos contables ya que con solo registrar los datos del cliente y el importe cobrado podríamos saber cuánto debemos pagar al fisco sin depender ciegamente de los tenedores de libros o de otra persona lo cual ocurre en la actualidad, siendo esto una desventaja para los TCP que no sabemos si la persona que nos realiza estos cálculos los hace correctamente o no y en caso de que no se haga correctamente el máximo responsable que es el TCP no se da cuenta hasta que le realizan una inspección o auditoría y es demasiado tarde. Sería muy productivo ese sistema ya que le permite al TCP tener una certeza razonable de que lo que está pagando mensual y anualmente es correcto y cumple con lo establecido en la legislación vigente.

Figura A10.13 Continuación entrevista #1

## ENTREVISTA

Fecha:

Nombre: Leandro  
Apellidos: Talledo Ramos

Actividad que realiza como TCP: Arrendador de Habitaciones

1. ¿Registra usted mismo a sus clientes en el Libro de Arredamiento?  SI  NO

1.1 En caso de que la respuesta sea NO especifique quién los registra?

Tenedor de libros  Familiar  Trabajador contratado

2. ¿Tiene conocimiento contable?  SI  NO

3. ¿Cumplimenta usted mismo su Declaración Jurada DJ-08 anualmente?  SI  NO

3.1 En caso de responder NO, especifique quién la cumplimenta?

Tenedor de libros  Familiar  Trabajador contratado

4. ¿Confecciona usted la contabilidad de su negocio?  SI  NO

4.1 En caso de que la respuesta sea NO, especifique quién la confecciona:

Tenedor de libros  Familiar  Trabajador contratado

5. ¿Utiliza un sistema contable?  SI  NO

5.1 En caso de que NO utilice un sistema contable responda lo siguiente:

5.1.1 ¿Qué lo limita a usar un sistema contable?

Es muy caro  Desconocimiento de cuáles existen  Otros

En caso de marcar la opción "Otros" especifique cuál es la otra limitante:

Creo que hay que saber algo de contabilidad y yo no se nada.

Figura A10.14 Entrevista #2

## ENTREVISTA

¿Qué aspecto negativo considera que trae consigo la limitante?

Que tengo que hacer todos los cálculos manualmente lo cual me ocupa mucho tiempo y es posible que los haga mal porque no tengo conocimiento contable.

6. ¿Le sería útil un sistema informático que le permita, a partir del registro de los clientes en un Libro de Arrendamiento digital, obtener el resultado del cálculo de los impuestos a pagar al fisco? ¿Por qué? ¿Considera que le traería algún beneficio?

Le sería muy útil al arrendador tener ese sistema ya que en la actualidad puede estar siendo subdeclarante sin saberlo por el desconocimiento lo cual no lo exonera de responsabilidad. Considero que los TCP necesitamos un sistema contable creado para nosotros, que permita realizar los registros contables, el cálculo de los impuestos, sin tener que tener mucho conocimiento contable y para en caso de que tenga un tenedor de libro que lo haga puede servirle de contrapartida y chequear así que su resultado es correcto. Además un sistema informático siempre es útil porque disminuye el tiempo de ejecución de los cálculos y disminuye la posibilidad de errores.

Figura A10.15 Continuación entrevista #2