	<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAr113</b>
	<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
	<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
		<b>PAGINA: 1 de 8</b>

21.1

<b>FECHA</b>	viernes, 5 de junio de 2020
--------------	-----------------------------

Señores  
**UNIVERSIDAD DE CUNDINAMARCA**  
 BIBLIOTECA  
 Ciudad

<b>UNIDAD REGIONAL</b>	Sede Fusagasugá
<b>TIPO DE DOCUMENTO</b>	Pasantía
<b>FACULTAD</b>	Ingeniería
<b>NIVEL ACADÉMICO DE FORMACIÓN O PROCESO</b>	Pregrado
<b>PROGRAMA ACADÉMICO</b>	Ingeniería de Sistemas

El Autor(Es):

<b>APELLIDOS COMPLETOS</b>	<b>NOMBRES COMPLETOS</b>	<b>No. DOCUMENTO DE IDENTIFICACIÓN</b>
Herrera Monroy	Luis Miguel	1.069.747.867

Director(Es) y/o Asesor(Es) del documento:

<b>APELLIDOS COMPLETOS</b>	<b>NOMBRES COMPLETOS</b>
Gutiérrez	Luis Alberto
Hernández Molano	Edward Norberto

Diagonal 18 No. 20-29 Fusagasugá – Cundinamarca  
 Teléfono (091) 8281483 Línea Gratuita 018000976000  
 www.ucundinamarca.edu.co E-mail: info@ucundinamarca.edu.co  
 NIT: 890.680.062-2

*Documento controlado por el Sistema de Gestión de la Calidad  
 Asegúrese que corresponde a la última versión consultando el Portal Institucional*



<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAr113</b>
<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
	<b>PAGINA: 2 de 8</b>

### TÍTULO DEL DOCUMENTO

INFORME FINAL DE PASANTÍA REALIZADA EN LA EMPRESA ICE COMPANY SAS PARA OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS.

### SUBTÍTULO

(Aplica solo para Tesis, Artículos Científicos, Disertaciones, Objetos Virtuales de Aprendizaje)

### TRABAJO PARA OPTAR AL TÍTULO DE:

Aplica para Tesis/Trabajo de Grado/Pasantía  
Ingeniero de Sistemas

### AÑO DE EDICION DEL DOCUMENTO

05/06/2020

### NÚMERO DE PÁGINAS

112

### DESCRIPTORES O PALABRAS CLAVES EN ESPAÑOL E INGLÉS (Usar 6 descriptores o palabras claves)

ESPAÑOL	INGLÉS
1. Reporte de errores y defectos (BUG)	Error and defect report (BUG)
2. Gestión de control de cambios (CCM)	Change Control Management (CCM)
3. Pruebas funcionales	Functional testing
4. Pruebas de regresión	Regression tests
5. Pruebas Automatizadas	Automated Testing
6. Aseguramiento de la calidad	Quality assurance (QA)



<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAr113</b>
<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
	<b>PAGINA: 3 de 8</b>

## RESUMEN DEL CONTENIDO EN ESPAÑOL E INGLÉS

(Máximo 250 palabras – 1530 caracteres, aplica para resumen en español):

### RESUMEN

El presente documento es el informe del desarrollo de la pasantía como modalidad de trabajo de grado para acceder al título de ingeniero de sistema, la pasantía consistió en la implementación y acompañamiento de procesos de calidad y testeos en el desarrollo de software a través de la metodología scrum en la empresa ice company sas, la intervención se basó en apoyar dicha empresa en acompañar procesos de testeos de funcionalidades desarrolladas en el diseño de un sistema POS (cabe resaltar que se crearon funcionalidades referentes a este, pero no se realizó un producto final debido a lo extenso y complejo de su desarrollo) en el sistema operativo Android, el cual va enfocado a establecimientos comerciales tipo restaurantes, se buscaba además que este proyecto tuviera procesos de calidad óptimos a través del manejo de herramientas que permitiera la fácil asignación de tareas, así como control de tiempos y seguimiento de errores encontrados.

### ABSTRACT

This document is the report of the development of the internship as a modality of Undergraduate work to access the title of System Engineer, the internship consisted of implementation and monitoring of quality processes and testing in software development through the scrum methodology in the ice company sas, the intervention was based on supporting said company to accompany test processes of functionalities developed in the design of a POS system (it should be noted that it will create functionalities references to it, but it will not be I make a final product due to the extensive and complex development) in the system Android operating, which is focused on commercial type restaurants, is It also sought that this project has optimal quality processes through the management of tools that allow easy assignment of tasks, as well as time control and tracking of errors found.



<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAR113</b>
<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
	<b>PAGINA: 4 de 8</b>

### AUTORIZACION DE PUBLICACIÓN

Por medio del presente escrito autorizo (Autorizamos) a la Universidad de Cundinamarca para que, en desarrollo de la presente licencia de uso parcial, pueda ejercer sobre mí (nuestra) obra las atribuciones que se indican a continuación, teniendo en cuenta que, en cualquier caso, la finalidad perseguida será facilitar, difundir y promover el aprendizaje, la enseñanza y la investigación.


En consecuencia, las atribuciones de usos temporales y parciales que por virtud de la presente licencia se autoriza a la Universidad de Cundinamarca, a los usuarios de la Biblioteca de la Universidad; así como a los usuarios de las redes, bases de datos y demás sitios web con los que la Universidad tenga perfeccionado una alianza, son:

Marque con una "X":

<b>AUTORIZO (AUTORIZAMOS)</b>	<b>SI</b>	<b>NO</b>
1. La reproducción por cualquier formato conocido o por conocer.	X	
2. La comunicación pública por cualquier procedimiento o medio físico o electrónico, así como su puesta a disposición en Internet.	X	
3. La inclusión en bases de datos y en sitios web sean éstos onerosos o gratuitos, existiendo con ellos previa alianza perfeccionada con la Universidad de Cundinamarca para efectos de satisfacer los fines previstos. En este evento, tales sitios y sus usuarios tendrán las mismas facultades que las aquí concedidas con las mismas limitaciones y condiciones.	X	
4. La inclusión en el Repositorio Institucional.	X	

De acuerdo con la naturaleza del uso concedido, la presente licencia parcial se otorga a título gratuito por el máximo tiempo legal colombiano, con el propósito de que en dicho lapso mi (nuestra) obra sea explotada en las condiciones aquí estipuladas y para los fines indicados, respetando siempre la titularidad de los derechos patrimoniales y morales correspondientes, de acuerdo con los usos honrados, de manera proporcional y justificada a la finalidad perseguida, sin ánimo de lucro ni de comercialización.

Para el caso de las Tesis, Trabajo de Grado o Pasantía, de manera complementaria, garantizo(garantizamos) en mi(nuestra) calidad de estudiante(s) y por ende autor(es) exclusivo(s), que la Tesis, Trabajo de Grado o Pasantía en cuestión, es producto de mi(nuestra) plena autoría, de mi(nuestro) esfuerzo personal intelectual, como consecuencia de mi(nuestra) creación original particular y, por tanto, soy(somos) el(los) único(s) titular(es) de la misma. Además, aseguro

	<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAR113</b>
	<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
	<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
		<b>PAGINA: 5 de 8</b>

(aseguramos) que no contiene citas, ni transcripciones de otras obras protegidas, por fuera de los límites autorizados por la ley, según los usos honrados, y en proporción a los fines previstos; ni tampoco contempla declaraciones difamatorias contra terceros; respetando el derecho a la imagen, intimidad, buen nombre y demás derechos constitucionales. Adicionalmente, manifiesto (manifestamos) que no se incluyeron expresiones contrarias al orden público ni a las buenas costumbres. En consecuencia, la responsabilidad directa en la elaboración, presentación, investigación y, en general, contenidos de la Tesis o Trabajo de Grado es de mí (nuestra) competencia exclusiva, eximiendo de toda responsabilidad a la Universidad de Cundinamarca por tales aspectos.

Sin perjuicio de los usos y atribuciones otorgadas en virtud de este documento, continuaré (continuaremos) conservando los correspondientes derechos patrimoniales sin modificación o restricción alguna, puesto que, de acuerdo con la legislación colombiana aplicable, el presente es un acuerdo jurídico que en ningún caso conlleva la enajenación de los derechos patrimoniales derivados del régimen del Derecho de Autor.

De conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, “*Los derechos morales sobre el trabajo son propiedad de los autores*”, los cuales son irrenunciables, imprescriptibles, inembargables e inalienables. En consecuencia, la Universidad de Cundinamarca está en la obligación de RESPETARLOS Y HACERLOS RESPETAR, para lo cual tomará las medidas correspondientes para garantizar su observancia.

**NOTA:** (Para Tesis, Trabajo de Grado o Pasantía):

**Información Confidencial:**


Esta Tesis, Trabajo de Grado o Pasantía, contiene información privilegiada, estratégica, secreta, confidencial y demás similar, o hace parte de la investigación que se adelanta y cuyos resultados finales no se han publicado. **SI X NO \_\_\_**

En caso afirmativo expresamente indicaré (indicaremos), en carta adjunta tal situación con el fin de que se mantenga la restricción de acceso.

**LICENCIA DE PUBLICACIÓN**

Como titular(es) del derecho de autor, confiero(erimos) a la Universidad de Cundinamarca una licencia no exclusiva, limitada y gratuita sobre la obra que se integrará en el Repositorio Institucional, que se ajusta a las siguientes características:

Diagonal 18 No. 20-29 Fusagasugá – Cundinamarca  
 Teléfono (091) 8281483 Línea Gratuita 018000976000  
 www.ucundinamarca.edu.co E-mail: info@ucundinamarca.edu.co  
 NIT: 890.680.062-2

	<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAR113</b>
	<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
	<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
		<b>PAGINA: 6 de 8</b>

a) Estará vigente a partir de la fecha de inclusión en el repositorio, por un plazo de 5 años, que serán prorrogables indefinidamente por el tiempo que dure el derecho patrimonial del autor. El autor podrá dar por terminada la licencia solicitándolo a la Universidad por escrito. (Para el caso de los Recursos Educativos Digitales, la Licencia de Publicación será permanente).

b) Autoriza a la Universidad de Cundinamarca a publicar la obra en formato y/o soporte digital, conociendo que, dado que se publica en Internet, por este hecho circula con un alcance mundial.

c) Los titulares aceptan que la autorización se hace a título gratuito, por lo tanto, renuncian a recibir beneficio alguno por la publicación, distribución, comunicación pública y cualquier otro uso que se haga en los términos de la presente licencia y de la licencia de uso con que se publica.

d) El(Los) Autor(es), garantizo(amos) que el documento en cuestión, es producto de mi(nuestra) plena autoría, de mi(nuestro) esfuerzo personal intelectual, como consecuencia de mi (nuestra) creación original particular y, por tanto, soy(somos) el(los) único(s) titular(es) de la misma. Además, aseguro(aseguramos) que no contiene citas, ni transcripciones de otras obras protegidas, por fuera de los límites autorizados por la ley, según los usos honrados, y en proporción a los fines previstos; ni tampoco contempla declaraciones difamatorias contra terceros; respetando el derecho a la imagen, intimidad, buen nombre y demás derechos constitucionales. Adicionalmente, manifiesto (manifestamos) que no se incluyeron expresiones contrarias al orden público ni a las buenas costumbres. En consecuencia, la responsabilidad directa en la elaboración, presentación, investigación y, en general, contenidos es de mí (nuestro) competencia exclusiva, eximiendo de toda responsabilidad a la Universidad de Cundinamarca por tales aspectos.


e) En todo caso la Universidad de Cundinamarca se compromete a indicar siempre la autoría incluyendo el nombre del autor y la fecha de publicación.

f) Los titulares autorizan a la Universidad para incluir la obra en los índices y buscadores que estimen necesarios para promover su difusión.

g) Los titulares aceptan que la Universidad de Cundinamarca pueda convertir el documento a cualquier medio o formato para propósitos de preservación digital.

h) Los titulares autorizan que la obra sea puesta a disposición del público en los términos autorizados en los literales anteriores bajo los límites definidos por la universidad en el "Manual del Repositorio Institucional AAAM003"

i) Para el caso de los Recursos Educativos Digitales producidos por la Oficina de Educación Virtual, sus contenidos de publicación se rigen bajo la Licencia Creative

	<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAr113</b>
	<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
	<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
		<b>PAGINA: 7 de 8</b>

Commons: Atribución- No comercial- Compartir Igual.



j) Para el caso de los Artículos Científicos y Revistas, sus contenidos se rigen bajo la Licencia Creative Commons Atribución- No comercial- Sin derivar.




**Nota:**

Si el documento se basa en un trabajo que ha sido patrocinado o apoyado por una entidad, con excepción de Universidad de Cundinamarca, los autores garantizan que se ha cumplido con los derechos y obligaciones requeridos por el respectivo contrato o acuerdo.

La obra que se integrará en el Repositorio Institucional, está en el(los) siguiente(s) archivo(s).

Nombre completo del Archivo Incluida su Extensión (Ej. PerezJuan2017.pdf)	Tipo de documento (ej. Texto, imagen, video, etc.)
1. INFORME FINAL DE PASANTÍA REALIZADA EN LA EMPRESA ICE COMPANY SAS PARA OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS	Texto e imágenes

En constancia de lo anterior, Firmo (amos) el presente documento:

APELLIDOS Y NOMBRES COMPLETOS	FIRMA (autógrafa)
Herrera Monroy Luis Miguel	

21.1-51-20

Diagonal 18 No. 20-29 Fusagasugá – Cundinamarca  
 Teléfono (091) 8281483 Línea Gratuita 018000976000  
 www.ucundinamarca.edu.co E-mail: info@ucundinamarca.edu.co  
 NIT: 890.680.062-2

*Documento controlado por el Sistema de Gestión de la Calidad  
 Asegúrese que corresponde a la última versión consultando el Portal Institucional*



<b>MACROPROCESO DE APOYO</b>	<b>CÓDIGO: AAAr113</b>
<b>PROCESO GESTIÓN APOYO ACADÉMICO</b>	<b>VERSIÓN: 3</b>
<b>DESCRIPCIÓN, AUTORIZACIÓN Y LICENCIA DEL REPOSITORIO INSTITUCIONAL</b>	<b>VIGENCIA: 2017-11-16</b>
	<b>PAGINA: 8 de 8</b>

Diagonal 18 No. 20-29 Fusagasugá – Cundinamarca  
Teléfono (091) 8281483 Línea Gratuita 018000976000  
[www.ucundinamarca.edu.co](http://www.ucundinamarca.edu.co) E-mail: [info@ucundinamarca.edu.co](mailto:info@ucundinamarca.edu.co)  
NIT: 890.680.062-2

*Documento controlado por el Sistema de Gestión de la Calidad  
Asegúrese que corresponde a la última versión consultando el Portal Institucional*



INFORME FINAL DE PASANTÍA REALIZADA EN LA EMPRESA ICE COMPANY SAS  
PARA OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS

LUIS MIGUEL HERRERA MONROY

UNIVERSIDAD DE CUNDINAMARCA

FACULTAD DE INGENIERÍA

INGENIERÍA DE SISTEMAS

Mayo de 2020

INFORME FINAL DE PASANTÍA REALIZADA EN LA EMPRESA ICE COMPANY SAS  
PARA OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS

LUIS MIGUEL HERRERA MONROY

Cod: 161214131

LUIS ALBERTO GUTIÉRREZ RAMÍREZ

Asesor metodológico

UNIVERSIDAD DE CUNDINAMARCA

FACULTAD DE INGENIERÍA

INGENIERÍA DE SISTEMAS

Mayo de 2020

## **Agradecimientos**

Agradezco a las personas quienes aportaron su tiempo para que este trabajo fuera posible, tanto las personas de la empresa ICE COMPANY SAS, el docente Luis Alberto Gutiérrez y el ingeniero Jorge Julio Reyes, por el apoyo y Retroalimentación de la información generada desde la universidad, de ellos valoro el conocimiento y compromiso con este proyecto. También a aquellas personas que estuvieron acompañándome en este proceso por disponer de su tiempo y paciencia para que este proyecto culminara satisfactoriamente.

## Tabla de contenido.

<b>Introducción</b> -----	<b>5</b>
<b>1. Justificación</b> -----	<b>6</b>
<b>Problema</b> -----	<b>7</b>
<b>2.1. Formulación del Problema</b> -----	<b>7</b>
<b>2.2. Planteamiento del Problema</b> -----	<b>7</b>
<b>3.Objetivos</b> -----	<b>9</b>
<b>3.1.Objetivo General</b> -----	<b>9</b>
<b>3.2.Objetivos Específicos</b> -----	<b>9</b>
<b>4.Fundamentación Teórico-Practica</b> -----	<b>10</b>
<b>5.Descripción de la Compañía</b> -----	<b>17</b>
<b>5.1.Nombre y descripción.</b> -----	<b>17</b>
<b>5.2. Sitio Web</b> -----	<b>17</b>
<b>5.3. Sector</b> -----	<b>17</b>
<b>5.4. Tamaño de la empresa</b> -----	<b>17</b>
<b>5.5. Especialidad</b> -----	<b>17</b>
<b>6.Descripción General</b> -----	<b>18</b>
<b>6.1.Recursos Utilizados</b> -----	<b>18</b>
<b>6.2.Metodología Aplicada</b> -----	<b>19</b>
<b>6.3Cronograma</b> -----	<b>25</b>
<b>7.Actividades Realizadas</b> -----	<b>27</b>
<b>7.1. Planificación de Pruebas</b> -----	<b>27</b>
<b>7.2 Recolección de la información.</b> -----	<b>28</b>

7.2 Sprint 1-----	28
7.3 Sprint. 2-----	28
7.4 Sprint 3-----	35
7.5 Sprint 4-----	40
7.6 Sprint 5-----	43
7.7 Sprint 6-----	45
7.8 Sprint 7-----	47
<b>8. Definición de pruebas</b>	
<b>9 Resultados -----</b>	<b>49</b>
<b>10. Conclusiones -----</b>	<b>51</b>
<b>11. Referencias-----</b>	<b>52</b>
<b>12. Tablas -----</b>	<b>53</b>
<b>13. Figuras-----</b>	<b>54</b>
<b>14. Apéndice -----</b>	<b>55</b>

## Introducción

El presente documento es el informe del desarrollo de la pasantía como modalidad de trabajo de grado para acceder al título de ingeniero de sistema, la pasantía consistió en la implementación y acompañamiento de procesos de calidad y testeos en el desarrollo de software a través de la metodología scrum en la empresa ice company sas, la intervención se basó en apoyar dicha empresa en acompañar procesos de testeos de funcionalidades desarrolladas en el diseño de un sistema POS (cabe resaltar que se crearon funcionalidades referentes a este, pero no se realizó un producto final debido a lo extenso y complejo de su desarrollo) en el sistema operativo Android, el cual va enfocado a establecimientos comerciales tipo restaurantes, se buscaba además que este proyecto tuviera procesos de calidad óptimos a través del manejo de herramientas que permitiera la fácil asignación de tareas, así como control de tiempos y seguimiento de errores encontrados.

La pasantía se apoyó también en la realización de pruebas a la plataforma DELY DOMICILIOS, la cual es una aplicación ya en etapa de producción que se encarga de gestionar domicilios entre clientes y establecimientos comerciales sirviendo como vitrina para que estos últimos presenten sus productos al público., estas intervenciones consistieron en la realización de pruebas de regresión, entre otras, que buscaban dar validación a actualizaciones que se generarán para esta plataforma, no solo en el entorno Android sino también en iOS.

La pasantía se llevó a cabo desde el 15 de septiembre de 2019 hasta el 07 de mayo de 2020. Tiempo en el cual se cumplió con las actividades dentro del marco del proyecto de pasantía.

### 3. Justificación

El testing es una de las fases más importantes en el ciclo de desarrollo de software, es tan importante que debería incluirse desde el inicio del proceso para identificar los errores que surjan en este no lleguen a acumularse en las fases finales de entrega, las empresas de software han de conceder un gran valor al área de calidad y testing ya que será esta la encargada de la identificación de errores cometidos en las etapas de desarrollo, además de que es la encargada de garantizar la calidad del producto, buscando así la satisfacción del cliente.

El proceso de calidad y testing de aplicaciones permite no solo la identificación de errores si no también garantiza la reducción de costos de mantenimiento ya que para determinado proyecto puede resultar muy costosa la corrección de errores en fases muy avanzadas de desarrollo del producto.

La presente pasantía concerniente al ejercicio de procesos de calidad y testeo busca indagar la pertinencia de las metodologías que un ingeniero usa al momento de realizar evaluación de funcionalidades y desempeño con el fin de que se puedan generar entregables más estables y limpios de errores.

Por otra parte, la pasantía es un espacio que ha permitir al ingeniero en formación, desarrollar sus habilidades en el manejo de proyectos de desarrollo de software, así como el uso de herramientas que permiten no solo la planificación sino también la organización de equipos de trabajo y gestión de tiempos de entrega.

## **Problema**

### **2.1. Formulación del Problema**

En cualquier proyecto de desarrollo de software es indispensable el manejo de etapas que permitan la verificación de resultados a través de diferentes tipos de pruebas, además de esto es importante que se lleve un control de cada actividad entregada.

En las metodologías tradicionales la gestión de pruebas de software se dejan para el final y usualmente dichas verificaciones producen que se generen represamientos en la entrega del producto final ya que al momento de validarse, surgen inconsistencias que hace que muchas etapas por las que tuvo que pasar el producto se vean afectadas, llegando a generar una reacción en cascada, estos efectos hacen que se aumenten los costes de recursos y se aumenten los tiempos planificados, pero ¿es posible optimizar el proceso de testing en un proyecto de software con el fin de reducir y mitigar riesgos en etapas avanzadas?.

Sin una planificación de pruebas, ni un proceso de calidad preciso, se puede llegar a un producto final defectuoso o que no cumpla con las expectativas que el cliente final buscaba, perdiendo tiempo es hacer reestructuraciones, cambios de último momento, disminuyendo la credibilidad frente al usuario final y llevándolo a tener una mala experiencia con este producto.

### **2.2. Planteamiento del Problema**

Con el fin de generar un proyecto que sea lo más limpio en cuanto a errores y fallos se refiere posible, la empresa ice company sas enfocada en el desarrollo de software, queriendo desarrollar las funciones iniciales para un sistema POS, busca reducir estas amenazas en su proyecto, implementando pruebas en cada etapa de dichas funcionalidades, para este fin solicita a través del acuerdo existente entre la universidad de cundinamarca y la empresa (acuerdo de



voluntades N° C-20180920-062) pasantes para esta área, ya que al ser compañía que cuenta con varios proyectos activos requiere apoyo para minimizar los riesgos que puedan surgir en estos.

Dentro de cada etapa planteada por la empresa se ha de buscar que cada proceso realizado cumpla con estándares de calidad, seguridad y se evalúen criterios de aceptación a través de diferentes pruebas de software, es aquí donde se lleva a cabo el proceso de pasantía, en donde el estudiante universitario mediante el conocimiento adquirido a través de su proceso formativo, apoya y acompaña la gestión de dichas actividades, proporcionando acciones que den solución a los requerimientos de la compañía.

### **3.Objetivos**

#### **3.1.Objetivo General**

Implementar y acompañar los procesos de calidad y testeo en el desarrollo de software a través de metodología scrum para la empresa ice company s.a.s

#### **3.2.Objetivos Específicos**

- evaluar los criterios de aceptación planteados para cada tarea de los proyectos intervenidos.
- asegurar la calidad de los procesos en el desarrollo de software de la empresa a través de herramientas de planificación de tareas y gestión de tiempos.
- Realizar diferentes tipos de pruebas de software buscando la detección de errores
- Enviar los fallos encontrados en la realización de pruebas al equipo de desarrollo para su verificación y corrección.

#### 4.Fundamentación Teórico-Practica

La implementación del término “Tester” tiene sus orígenes en el año 1947 cuando el equipo encargado de trabajar en la súper-computadora “Mark II” de la universidad de harvard compartió el primer reporte de un error, causado nada más y nada menos que por algo que ellos denominaron “bicho” lo que se conoce en el mundo informático como “bug”.

El fallo que presentó esta súper-computadora consiste en un simple insecto, una polilla, la cual se había colocado en un relé electromagnético y que provocaba que este quedara abierto y no funcionará correctamente.

Para el reporte de esta anomalía se nombró a esta polilla como “ bicho” lo cual dio posteriormente origen al término “bug” para referirse a un error o fallo en el software o en algún punto del sistema que puede llegar a desencadenar un resultado inesperado.

Esta historia hace parte del origen de una nueva área de TI que con el tiempo fue madurando su conceptualización y tomando más peso hasta lo que hoy se conoce como “pruebas de software”.

Si una aplicación fuese lanzada sin ser probada, esta podría no llegar a tener el funcionamiento correcto, lo que conlleva a un impacto negativo para los usuarios y esto implicaría una pérdida de dinero, tiempo e imagen corporativa, es por esto que cada día cobra más importancia la necesidad de realización de pruebas para velar por la calidad de los sistemas y del software. Pero entonces ¿Cuál es la función de un tester?. Los testers son los encargados de velar por la calidad del software, son los que garantizan que todo funcione correctamente de acuerdo a lo definido y que los defectos encontrados no lleguen al usuario. Hoy día el desarrollo de software no se concibe sin la realización de pruebas ya que estas son las que darán garantía para el aseguramiento de la calidad de dicho software.

## Figura 1

*Diferenciación entre error, defecto y fallo.*



Nota. Se entiende que un error producido por un desarrollador causa un defecto en el software al momento de ser ejecutado puede provocar fallos en el sistema. Pero los fallos que afectan el sistema también pueden ser causados por agentes externos. Mediante el testing, se localizan estos fallos causados por los defectos, y por lo general es el desarrollador quien se encarga de esta tarea y de corregir dicho defecto.

### **Cualidades y habilidades del Tester**

#### ***Curiosidad.***

Un buen tester debe tener amplios conocimientos de lo que va a probar, pero también debe ser curioso y deberá plantear en cada situación que sucede.

#### ***Observación***

un buen tester tiende a analizar detalles que los demás no ven y prioriza cosas que otros no ven con mucha importancia

#### ***Razonamiento Creativo***

Consiste en buscar y resolver problemas por diferentes maneras y vías alternativas, en ocasiones, de manera creativa e imaginativa.

### ***Comunicación***

Debe tener la capacidad de transmitir las incidencias encontradas y reportarlas de una manera clara y concisa ya sea de manera oral o escrita, buscando generar ideas concretas y de valor.

### ***Empatía***

a la hora de reportar fallos, el tester debe entender que está en el mismo barco con el equipo de desarrollo y debe darlo a entender de esa manera, y que lo más fundamental para ambos equipos es la calidad del software.

### **Roles y Perfil Dentro del Testing**

Existen diferentes tipos de roles y perfiles y son utilizados según el tipo de proyecto y del tipo de organización los más comunes y utilizados son:

**QA Manager**, orientado al control y gestión de las pruebas

**Analista QA**, perfil encargado generalmente del diseño de las pruebas

**Tester**, persona encargada de ejecutar los casos de prueba

**Profesionales con Perfiles Más Técnicos**, encargados de la automatización de las pruebas o de pruebas de rendimiento

### **La Evolución de las Pruebas de Software**

Hasta hace muy poco las pruebas de software eran dominio exclusivo de las áreas de TI operando bajo el siguiente modelo:

- **Las pruebas corresponden a la última fase del ciclo de desarrollo**, previo a la salida en producción, buscando probar si el software desarrollado tenía errores. Esto daba origen a retrocesos, incremento en costos e incumplimiento en tiempos de entrega.
- **Se hacían pruebas Ad hoc**, pensadas únicamente para una etapa o situación específica, dejando por fuera un sinnúmero de variables posibles.
- **Sólo se probaba las peticiones del cliente** sin indagar sobre los por qué dejando de lado la esencia del Testing: objetividad, independencia y criterio.
- **El modelo de cobro del servicio** era basado casi exclusivamente en el número de horas invertidas.
- **Las pruebas eran consideradas un sobre costo** en el que se debía incurrir para cerrar el ciclo de desarrollo sin la conciencia del aporte en la disminución de riesgos, prevención de reprocesos, incremento de velocidad y oportunidad en la entrega a producción.
- **Los analistas de pruebas no eran valorados profesionalmente** por no considerarse el Testing una disciplina estudiada sino una práctica a la que se llegaba accidentalmente, adicionalmente no tenía el requisito de saber desarrollar software.

Cada día las pruebas de software se desplazan más hacia las etapas iniciales del del ciclo de desarrollo cambiando la forma de operar hasta entonces, esto es:

- **Las pruebas inician desde etapas tempranas** del desarrollo.
- **Amplitud en los tipos de pruebas** centrados en la calidad de los aplicativos.
- **El número de errores detectados** determina la efectividad de la inversión en pruebas y no el tiempo invertido.
- **Se impone la automatización básica** de algunos elementos básicos y simples.

- **Las pruebas son consideradas valiosas** para el crecimiento de las empresas pues evitan que los usuarios tengan reprocesos perdiendo oportunidad e incrementando los costos del proyecto.

### **El Uso de Metodologías Ágiles Para el Aseguramiento de la Calidad**

A pesar de que cada vez son más las empresas que apuestan por el uso de metodologías ágiles en sus equipos de desarrollo de software , aún son muchas las que desconocen la flexibilidad, agilidad y ahorro de tiempo que supone el uso de estos procedimientos.

La llegada de metodologías ágiles como Scrum, permite abordar de una manera ágil y rápida los procesos que conforman la vida de un proyecto, llevando a cabo desarrollos por etapas o sprints, realizando subidas a producción más cortas, incorporando herramientas de integración continua, y fomentando la comunicación directa con el cliente para contrastar los resultados de manera periódica.

El objetivo principal no es otro más que la detección de errores en fases tempranas para rectificar en cualquier etapa previa a la subida a producción, pero sin lugar a dudas la mayor ventaja aportada por las metodologías ágiles es la creación de productos de calidad que logren la satisfacción del cliente final, esto a dado gran visibilidad a la importancia de la labor que realizar los responsables de calidad de software

La responsabilidad del QA no solo tiene lugar tras la fase de desarrollo, sino que está presente durante todo el proceso , validando la aptitud del producto en cada una de sus etapas.

### **El Proceso de Pruebas**

el proceso de pruebas llevado a cabo por el equipo de QA a lo largo de un proyecto no solo dará lugar a un producto fiable y de calidad si no que permitirá además, establecer un

control permanente sobre todos los procesos, evitando que posibles problemas aparezcan en fases más avanzadas.

Dentro de las características principales del equipo de Qa Esta la planificación de sus actividades en cada una de las fases de un proyecto

### ***Fase de Análisis***

En esta etapa se verifican que los objetivos y requisitos planteados por el cliente sean correctos y alcanzables y con esto se elabora un plan de pruebas.

### ***Fase de Diseño y Desarrollo***

En esta fase el equipo de QA será el encargado de velar por el cumplimiento de cada uno de los requisitos y criterios de aceptación, vigilando que la solución se adapte a la propuesta funcional y de diseño

Para esta fase se busca automatizar procesos de compilación, se revisa el código generado y se realizan entregas continuas conforme van siendo desarrolladas las tareas planteadas.

### ***Fase de Validación y Entrega***

En esta fase se ejecutará el plan de pruebas, revisando que las soluciones ya desarrolladas respetan los requisitos fijados y la entrega al cliente se hace en condiciones adecuadas, en este caso se entregan reportes e informes de nuestro plan de pruebas.

Todos estos procedimientos se llevarán a cabo cada sprint con el fin de manejar un proceso iterativo que es lo que busca siempre las metodologías ágiles.

### **Tipos de Pruebas de Software**

Todos los tipos de pruebas se pueden definir básicamente en dos grupos: pruebas funcionales y pruebas no funcionales

Dentro de las pruebas funcionales tenemos



**Pruebas unitaria**, es una forma de comprobar el correcto funcionamiento de una unidad de código.

**Pruebas de Aceptación**, estas son las pruebas finales que se realizan , en estas, el cliente prueba el software y verifica que cumpla con los objetivos planteados y sus expectativas

**Pruebas de Integración**, este tipo de pruebas se busca saber si la comunicación entre los distintos componentes presentes en el software es funcional

**Pruebas de Regresión**, aquí se está comprobando que el código que se ha modificado se comporte como se quiere, y que él los cambios efectuados no hayan causado otro tipo de problemas en otras partes del software.

Dentro de las pruebas no funcionales se pueden destacar:

**Pruebas de Carga**, estas pruebas consisten en simular una carga de trabajo similar y superior al que tendrá el software cuando esté en funcionamiento, con el fin de observar el nivel de de usuarios que soporta el sistema en determinadas concurrencia de usuario.

**Pruebas de Estrés**, consiste en sobrecargar un sistema más allá de sus límites para saber como y cuando fallara.

**Pruebas de Portabilidad** es el proceso de verificar el grado de facilidad o dificultad con que un sistema puede ser efectiva y eficiente transfiriendo a otro entorno ya sea otro hardware u otra plataforma de sistema operativo.

## **5.Descripción de la Compañía**

### **5.1.Nombre y descripción.**

ICE COMPANY SAS es una empresa fundada en 2013 dedicada al desarrollo e implementación de software pos para cualquier tipo de negocio en las diferentes áreas comerciales, ofrece soluciones integrales diseñadas especialmente para las empresa, posee capacidad integración de todas sus soluciones de software, ofrece también desarrollo de todo tipo de aplicaciones comerciales, su financiación es 100% capital privado, actualmente genera alrededor de cinco(6) empleos directos y más de veinte(20) empleos de manera indirecta.

### **5.2. Sitio Web**

<http://www.icecompany.com.co>

### **5.3. Sector**

Desarrollo de programación

### **5.4. Tamaño de la empresa**

6 empleados

### **5.5. Especialidad**

Desarrollo de software, Software para punto de venta, Software para control de inventarios, Desarrollo a la medida.

## **6.Descripción General**

### **6.1.Recursos Utilizados**

Para la correcta realización de las actividades propuestas, la empresa ha puesto a disposición un número de plataformas y aplicaciones que permitiendo solo la comunicación directa entre personas involucradas en los proyectos, sino que también ha dispuesto de repositorios para poder dar continuidad a avances que se realizan en los proyectos implicados.

Los recursos puestos a disposición son:

#### ***AppCenter***

Microsoft Apps Center se presenta como el centro de control de todas las aplicaciones móviles. Permite transferir todas las apps, en cualquiera de sus lenguajes, a los servicios de la nube para obtener ciclos de lanzamiento más rápidos, aplicaciones de mayor calidad y ofrecer todos los datos necesarios para mejorar la experiencia de usuario.

#### ***Slack***

La herramienta está enfocada en crear un espacio de trabajo digital donde se hace más eficiente la comunicación e intercambio de tareas dentro de las organizaciones. La plataforma permite a cualquier equipo de la organización tener un chat room donde los miembros de un equipo o proyecto en cuestión se reparten las tareas, se envían avances y se pueden comunicar de forma rápida.

#### ***Horde Webmail***

Es un gestor de correo web que cumple todas las funciones típicas de un servicio de correo electrónico. Esta aplicación incluye herramientas como calendario, libreta de direcciones, listas de tareas, bloc de notas y más.

### ***Jira***

Es una herramienta en línea para la administración de tareas de un proyecto, el seguimiento de errores e incidencias y para la gestión operativa de proyectos. La herramienta fue desarrollada por la empresa australiana Atlassian. Dispone también de paneles de control adaptables, filtros de búsqueda, estadísticas y función de correo electrónico.

### ***Postman***

es una plataforma de desarrollo de APIs que se basa por un modelo de desarrollo API First. nace como una herramienta que principalmente nos permite crear peticiones sobre APIs de una forma muy sencilla y poder probar su correcto funcionamiento.

### ***Android Studio***

Es un IDE, una interfaz de desarrollo. En realidad, es una especie de escritorio de trabajo para un desarrollador. Allí se encuentra nuestro proyecto, las carpetas del mismo, los archivos que hay en él, y todo lo necesario para acabar creando la aplicación.

### ***BlueStacks App Player***

Es una herramienta que permite ejecutar aplicaciones Android en nuestro PC a través de una cómoda interfaz. Una vez que accedemos podremos descargar apps como lo haríamos en nuestro propio dispositivo y trabajar o jugar con ellas sin problemas de compatibilidad.

## **6.2. Metodología Aplicada**

Este proyecto tuvo la utilización de metodologías ágiles como lo es SCRUM, esta metodología fue la implementada ya que adopta estrategias de desarrollo incremental en lugar de planificar y ejecutar completamente el resultado, SCRUM es un modelo que puntualiza un conjunto de prácticas y roles que llevan a una planificación de tareas con duración entre una y cuatro semanas(Sprint), tiempo en el cual el equipo de trabajo crea avances que son

potencialmente funcionales. El conjunto de características que forma parte de cada sprint viene del Product Backlog, que es un conjunto de requisitos de alto nivel priorizados que definen el trabajo a realizar.

Esta metodología fue la ideal para el desarrollo del proyecto ya que es fácil manejarlo a manera de un cronograma de actividades, pudiendo dividir dicho proyecto en pequeños bloques o Sprints con el fin de que se fuera revisando y mejorando la fase anterior, esta metodología permite además que los equipos generen una auto organización, y se centralizaron en en la calidad del producto final.

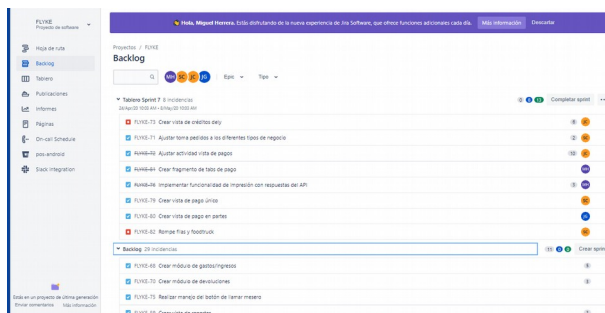
### ***Implementación de la metodología en el proceso de pasantía***

Durante el proceso de desarrollo y testeo de las funcionalidades planteadas para el sistema POS llamado “FLYKE” se trabajó de la siguiente manera, cumpliendo con lo conceptos planteados por esta metodología.

De manera inicial se contaba con una lista de requerimientos y funcionalidades que debería cumplir el sistema a desarrollar a esta lista se le conoce como Backlog. Estos requerimientos fueron diseñados inicialmente por el product owner, quien priorizar las tareas.

## **Figura 2**

*Backlog del proyecto FLYKE*



*Nota.* La imagen visualiza el backlog del proyecto POS a través de la herramienta JIRA la cual permite hacer un seguimiento de actividades, tiempos de entrega, asignación de roles, generación de informes entre otras cosas.

Posteriormente estos requerimientos se dividieron en tareas las cuales eran asignadas a los diferentes miembros del equipo, los cuales puntuaron un tiempo estimado para su desarrollo y de la misma manera se plantaba una estimación de tiempo para la comprobación de la tarea y sus respectivas pruebas.

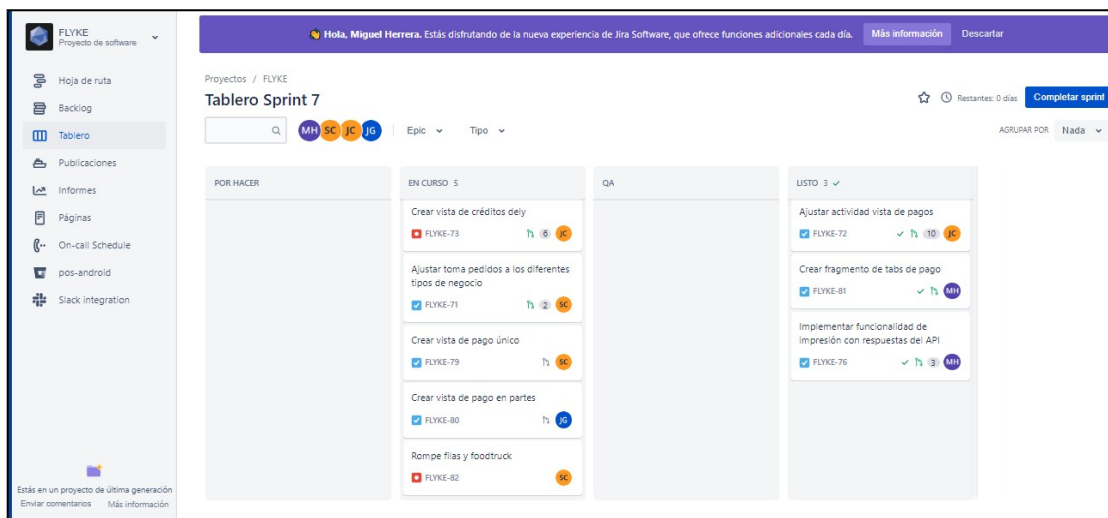
Después de priorizar las tareas del sprint a través del sprint planning la cual es una reunión que se realiza al inicio de cada ciclo para determinar duración de dicho sprint, las tareas que contempla, asignación de estas, se realizó el desarrollo de estas.

Estos sprints tuvieron duraciones que fueron desde 2 a 4 semanas, contando con un total de ocho(8) sprints a lo largo del trabajo realizado en la pasantía. Algunos más cortos que otros debido a la priorización de tareas.

El seguimiento de cada sprint se realizó a través de la plataforma de atlassian con su módulo JIRA la cual permite planificar el product backlog así como organizar las tareas por sprints, asignación de roles y estimación de tiempos de entrega.

### **Figura 3**

*Tablero de actividades de uno de los sprint visualizado en JIRA*



A través de esta herramienta fue posible puntuar también mediante puntos de historia ( cada punto de historia equivale a 4 horas de trabajo), el tiempo estimado de verificación cada tarea evaluada.

Manejando tareas a través del tablero ágil de cada sprint se buscó dar un flujo de control a dichas tareas, ubicadas inicialmente en el estado “por hacer” allí eran asignadas a los miembros del equipo de desarrollo el cual daba una puntuación estimada a cada tarea, posteriormente cuando se empezaba a trabajar en estas eran movidas al estado “en curso”, cuando se daba por desarrollada, esta era enviada al estado de “QA” en donde se realizaba la verificación de los criterios de aceptación propuestos por el cliente, allí se daba una nueva estimación para la realización de dichas pruebas, si el trabajo realizado en la tareas no cumplía con los requerimientos era devuelta a un estado “en curso” para su corrección y se había de crear una nueva incidencia que visualiza la ubicación del problema a encontrado y posteriormente este error era notificado a el desarrollador inicial.

#### **Figura 4**

*Criteria de aceptación de una tarea en el proyecto FLYKE*



*Nota.* Se puede visualizar que la herramienta permite asignar roles según la etapa en la que se encuentre la tarea, además tiempos estimados de entrega, adición de comentarios, incluir archivos multimedia a los requerimientos, entre otras cosas.

### ***Roles de la Metodología Encontrados en la Pasantía***

***Product Owner.*** Es la persona que se encargó de generar toda la lista de requisitos y la priorización de tareas, en la compañía este cargo estuvo bajo la responsabilidad del jefe de tecnología (CTO), el cual también era el encargado de definir si un sprint era válido o si no, para nuestro caso el product owner viene siendo la referencia de cliente.

***Scrum Master.*** Este rol se especializó en revisar que toda la metodología se cumpliera, además de mantener la conexión entre el equipo de desarrollo y el product owner, gestionar las reuniones del grupo de trabajo y velar por la evolución del tablero de actividades, verificando además que los miembros del equipo estuvieran siempre entregando nuevas avances. Este cargo estuvo bajo el mando del jefe ejecutivo (CEO) de la compañía.

***Equipo de Desarrollo y Equipo de QA.*** Conformando el primer equipo por un total de cuatro (4) miembros los cuales se encargaron de desarrollar la tareas que posteriormente eran evaluadas por el equipo de QA conformado por 2 miembros quienes realizamos las evaluaciones y validaciones a dichas tareas programadas.



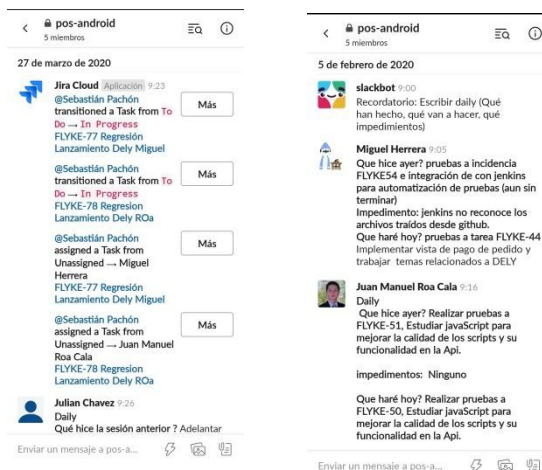
Todos los días antes de iniciar actividades, a través de la herramienta Slack se generó el “daily planning” el cual consiste en una rápida revisión por parte de cada integrante en donde se digitaba a través del chat de la herramienta, que se hizo la sesión anterior, que haría la sesión de ese día y que impedimentos tiene que no le permiten avanzar con la tarea asignada, esto contribuyó a que el equipo no se quedaría estancado en tareas sin desarrollar y con el apoyo de los demás colaboradores se dio avance a la tarea.

Esta herramienta enlazada con la suite de atlassian permitió además visualizar a través de notificaciones al chat del equipo cualquier cambio en el estado de una actividad al tablero de actividades y los cambios al código al momento de hacer una integración nueva al proyecto

También se hizo uso del correo corporativo de la empresa el cual permite el envío de archivos un poco más pesados de ser requerido, y un canal secundario para la comunicación entre el equipo de trabajo, este también sirvió de enlace entre las diferentes plataformas que requieren un inicio de sesión, dando un parte de responsabilidad por parte de la empresa buscando que sus vinculados tuvieran una identificación dentro de la compañía.

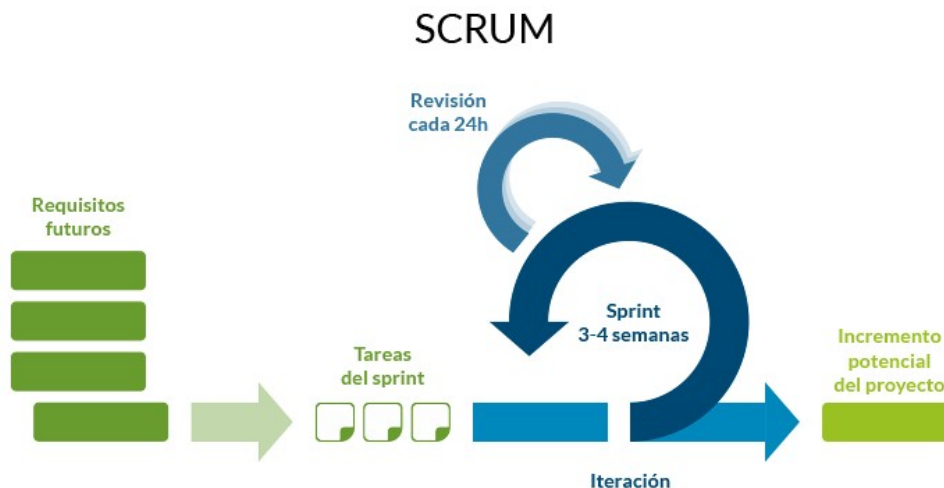
## Figura 5

*Uso de la herramienta Slack para el control del daily-planning y notificaciones de cambios en el proyecto.*



**Figura 6**

*Proceso realizado en cada uno de los sprint*



*Nota.* En el gráfico se observa el proceso por el cual se rige cada uno de los sprint realizados en el proyecto, empezando por una obtención de los requerimientos, de estos se desglosan las tareas que harán parte del sprint el cual puede tener una duración entre 3 y 4 semanas dependiendo de la priorización de las tareas, al final de este se realiza una reunión de retrospectiva donde se analiza lo bueno, y lo malo del sprint y se realizan los entregables para posteriormente en el sprint planning realizar nuevamente obtención de requerimientos para el siguiente sprint, también se realizan reuniones diarias no mayor a 15 minutos, para caso del proyecto de pasantía esto se reemplazó por el uso de la herramienta slack para notificar los avances diarios.

### **6.3Cronograma**

la siguiente tabla representa las principales tareas y funcionalidades en cada uno de los sprint propuestos en referencia al tiempo estimado de verificación y evaluación de pruebas.

Actividades y funcionalidades	Sprint0				Sprint1				Sprint2				Sprint3				Sprint4				Sprint5				Sprint6				Sprint7			
	semana				semana				semana				semana				semana				semana				semana							
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Planificación de pruebas	■	■																														
Recopilación de información			■	■																												
Inicio de sesión					■					■				■								■										
gestión de módulos						■				■								■	■			■										
Gestión de productos							■				■	■		■																		
Modulo de facturación															■			■					■	■		■						■
Modulo de empleados								■	■					■	■	■			■								■					
Modulo de cierre de caja																			■				■							■		
Modulo de toma pedidos														■				■	■				■			■				■		
Gestión de impresiones																							■				■				■	
Modulo de ajustes																			■													
Proyecto dely		■	■																							■	■	■				

**Tabla 1**

*Cronograma de Actividades*

## 7.Actividades Realizadas

### 7.1. Planificación de Pruebas

Dentro de la planificación de pruebas se tuvo en cuenta la metodología seleccionada para el proyecto, y se tuvieron en cuenta los siguientes criterios:

- El equipo de testing deberá estar integrado con el equipo de desarrollo con el fin de que exista una comunicación más directa
- El testing debe ser permanente, incluso cuando el equipo de desarrollo no ha terminado de integrar el producto
- La única forma de lograr la entrega de una tarea es realizando las pruebas pertinentes, esto implica que, cada código al ser desarrollado debe pasar a revisión para posteriormente ser integrado al proyecto base.
- Al integrarse nuevo código se debe realizar pruebas buscando que dicha integración no afecte otras funcionalidades del proyecto. (regresión).

Para el momento de la planificación es importante conocer el alcance de cada una de las pruebas realizadas, así como el tipo de prueba que se aplique según el caso requerido, los criterios de aceptación y otros aspectos como la estimación de tiempos, recursos que harán parte del proceso y la preparación del entorno.

Dentro de esta planificación se realiza el plan de pruebas el cual determina los conceptos de cada una de las pruebas realizadas así como el procedimiento a seguir al momento de la ejecución de estas, así como los pasos a seguir para el reporte y seguimiento de las diferentes intervenciones por parte del pasante en el área de testing. (*sección de anexos*).

### 7.2 Recolección de la información.

Una vez elaborado el plan de pruebas se procedió al análisis de toda la documentación y recopilación de la información necesaria con respecto al sistema (FLYKE), con la ayuda del

SCRUM MASTER se logra evidenciar el alcance del proyecto, la arquitectura utilizada el diseño de las interfaces de usuario preliminares, así como los datos de prueba necesarios.

Estos dos procesos (planificación y recolección ) fueron parte del Sprint 0 dando pauta al inicio del proyecto FLYKE el cual busca desarrollar un sistema POS para dispositivos Android (tabletas) que supla las necesidades de establecimientos comerciales en el manejo de facturas, gestión de empleados inventarios, entre otros.

## **7.2 Sprint 1**

Para este proceso se tomaron los criterios de aceptación estipulados por el jefe de proyecto y fueron validados a través de pruebas de interfaces y seguridad, observando el comportamiento en los campos de registro, cuidando que todos los campos tengan su respectivo validar (que el campo de correo sea tipo correo, el campo de contraseña mantenga un ocultamiento de la misma). Además de verificar que las vistas desarrolladas cumplieran los criterios teniendo en cuenta las interfaces de guía.

Cabe resaltar que algunas actividades al no ser terminadas en su totalidad durante el sprint asignado o cuando su evolución dependa de otro tipo de actividad, estas pasan al siguiente sprint con el fin de darles continuidad.

A continuación se visualiza cada una de las tareas pertenecientes al sprint 1 definiendo , responsables de cada actividad, tiempo estimado de desarrollo y de testeo, así como sus criterios de aceptación. Validando su finalización después de realizadas las pruebas correspondientes.

### **Tabla 2**

*Información de la actividad FLYKE-34*

<b>[FLYKE-34] Crear la interfaz de inicio de sesión</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Ivan Quintero Torres		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 1, Tablero Sprint 2		
<b>Story point estimate:</b>	4		
<b>Qa Story point estimate:</b>	4		
<b>Descripción</b>	<p>Se requiere que se genere la interfaz de inicio de sesión que corresponderá al inicio único de sesión de la aplicación Flyke</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Seguir el diseño de inicio de sesión.</li> <li>● Se debe tener dos campos, uno de correo, y uno de contraseña. Se debe validar que el campo de correo sea de tipo correo. Y el campo de contraseña sea oculto.</li> <li>● Se debe validar en inicio de sesión el largo de la contraseña y que se muestren errores concordantes.</li> <li>● Se debe en registro validar la existencia de los campos requeridos y el largo de la contraseña y que la confirmación sean iguales</li> </ul>		

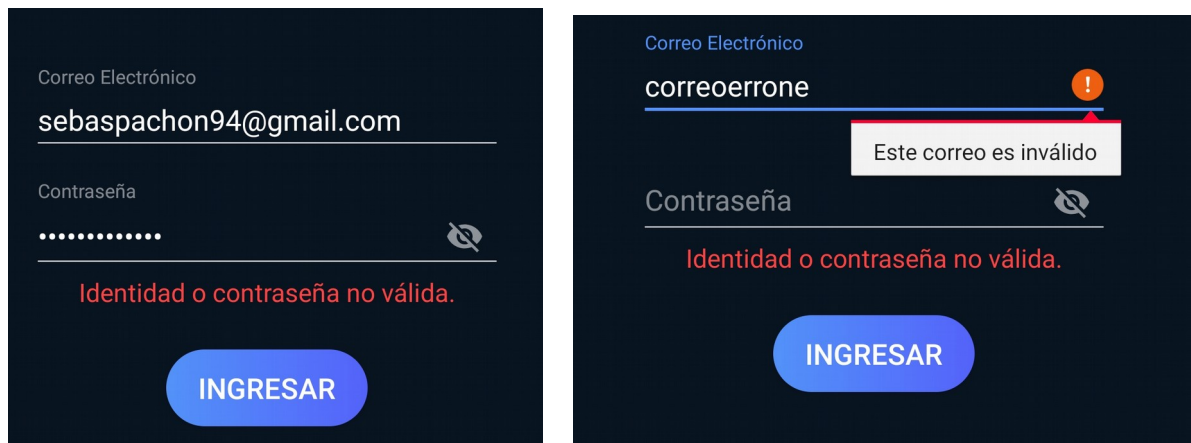
*Nota.* En la tabla se puede observar el tipo de actividad a realizar así como los involucrados en el cumplimiento de la misma, tiempos de realización estimados, ruta de ubicación según el sprint asignado, los criterios de aceptación que debe cumplir la tarea para ser válida y el nivel de prioridad que posee.

### ***Ejecución de pruebas para FLYKE-34***

Para la validación de esta actividad se realizaron pruebas de seguridad, verificando que cada campo en el inicio de sesión cumpliera con los parámetros establecidos, además de pruebas de interfaz buscando que se cumpliera la visualización conforme al diseño.

**Tabla 3***Prueba de seguridad tarea FLYKE-34*

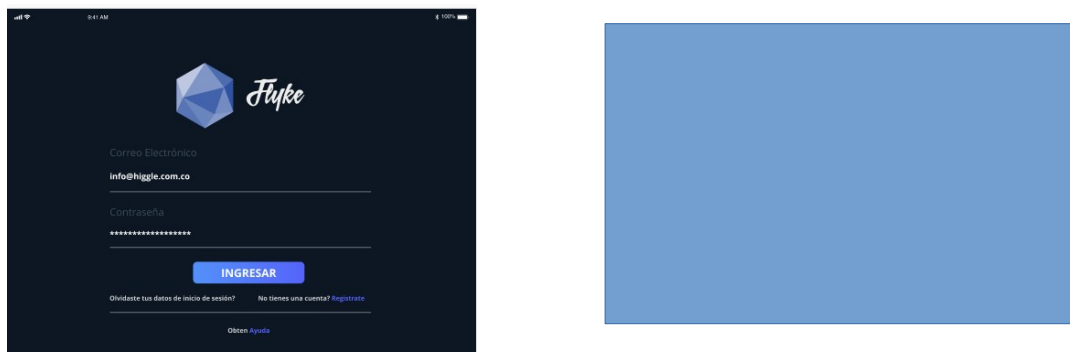
Tipo de prueba realizado	Prueba de seguridad
Pasos de la Prueba a realizar	Ingresar datos validos y erróneos en cada campo del formulario de inicio de sesión observar si se valida el tipo de datos ingresados( en correo sea tipo correo, en contraseña tenga ocultamiento de caracteres.) validar que los campos estén completos
Resultados Esperados	Se debe validar correctamente los datos, de ser erróneos que se genere un mensaje de alerta, mostrando al cliente el tipo de error. De la misma manera el formulario de registro de cuenta debe cumplir con parámetros como largo mínimo de caracteres en contraseña y que los datos ingresados sean validados según el tipo de dato.
Resultados Reales	Al ingresar datos erróneos el sistema de login le genera al usuario una alerta con el mensaje “este correo es invalido”. Y al momento de ingresar contraseña errónea muestra el mensaje “identidad o contraseña no valida.” sin embargo aun se encuentra inconclusa debido a falta de interfaces de registro solo es posible validar el formulario de inicio de sesión.

**Figura 7***Verificación de campos en el inicio de sesión***Tabla 4***Prueba de interfaz tarea FLYKE-34*

Tipo de prueba realizado	Prueba de interfaz
Recurso utilizado	Adobe XD
Pasos de prueba a realizar	Comprobación de vistas diseñadas vs interfaces creadas.
Datos de la prueba	Vistas prediseñadas
Resultados esperados	Las interfaces creadas en esta tarea sean lo mas similares posibles a las diseñadas previamente, cumpliendo con la misma posición de botones, campos de texto, tanto la interfaz de registro como la interfaz de inicio de sesión
Resultados reales	La interfaz de inicio de sesión cumple con las características y criterios pero la interfaz de registro aun se encuentra en desarrollo por ende no pudo ser evaluada. Y pasa a ser parte del segundo sprint

## Figura 8

*comparativa interfaz preliminar vs interfaz desarrollada*



*nota. La imagen a la izquierda corresponde a las vista creada en adobe XD y la de la derecha es la interfaz final del funcionamiento de inicio de sesión.*

## Tabla 5.

*Información de la actividad FLYKE-35*

<b>[FLYKE-35] Crear la interfaz de selección de categoría</b>	
<b>Estado:</b>	Finalizada
<b>Proyecto:</b>	FLYKE



<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Juan David Gutiérrez		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 1		
<b>Story point estimate:</b>	2		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>Al iniciar sesión se debe redirigir a la interfaz de selección de categoría. Debe salir un listado cuadrulado de las categorías disponibles.          Debe hacerse con la ayuda un recyclerview y un adaptador.          (Dinámica)          Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Se debe llegar después de hacer inicio de sesión.</li> <li>● Se debe mostrar una lista de categorías dummy, que permitan visualizarse como una cuadrícula.</li> <li>● Al dar click en una categoría debe redirigirlo a la interfaz de productos de una categoría</li> </ul>		

Lo que se busco en esta actividad fue una continuidad y una transición entre diferentes vistas, yendo desde el inicio de sesión hasta la vista de categorías de sistemas, además de validar su diseño y que se cumpliera con los criterios de aceptación.

### ***Ejecución de pruebas para FLYKE-35***

*En esta tarea se realizo pruebas de usabilidad y de interfaces validando así su continuidad entre vistas, así como su similitud con la interfaz diseñada para la funcionalidad cuidando que sus componentes estén ubicados de manera similar al diseño preliminar.*

### **Tabla 6**

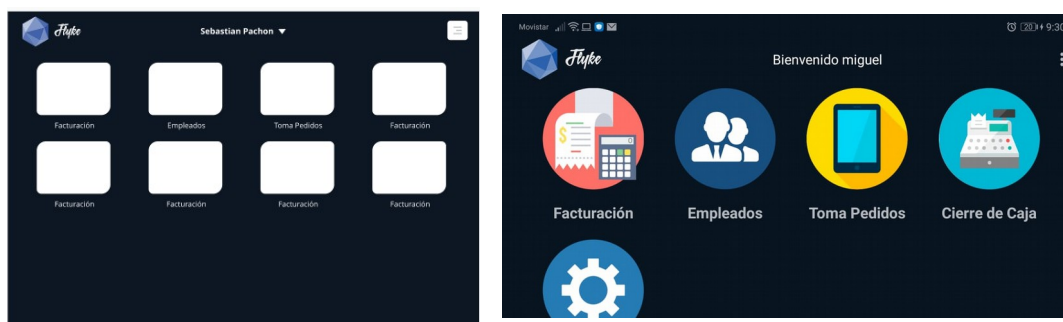
#### *Prueba de interfaz FLYKE-35*

Tipo de prueba realizado	Prueba de interfaz
--------------------------	--------------------

Pasos de la Prueba a realizar	Comparativa entre vistas diseñadas en adobe XD vs las interfaz desarrollada
Resultados Esperados	Las vistas desarrolladas dentro de la actividad corresponden al mismo diseño de las que se planearon en el diseño de interfaces, logrando la ubicación de los diferentes elementos de esta.
Resultados Reales	Las interfaces son similares, sin embargo se diseño los cajones de cada modulo de la categoría de manera ovalada, en el diseño preliminar se encontraban de manera rectangular. La forma que fue diseñada no afecta en nada la funcionalidad de los procesos y aporta una visualización mas agradable para el usuario. Por tanto se da por aceptada con el habla del product owner.

### Figura 9

*vista de categorías. Interfaz diseñada vs interfaz implementada.*



*Nota.* A la izquierda, la interfaz diseñada en adobe XD, a la derecha, la interfaz implementada.

### Tabla 7

#### *Prueba de usabilidad FLYKE-35*

Tipo de prueba realizado	Prueba de usabilidad
Pasos de la Prueba a realizar	Iniciar sesión, verificar la transición de vistas, visualizar vista de categorías
Resultados Esperados	La transición entre vistas desde inicio de sesión debe ser completa, manejando una velocidad de una actividad a la otra sin retrasos, y que se logre identificar cada uno de los módulos iniciales del sistema desarrollado.
Resultados Reales	La transición es buena, cumple con las expectativas, además que se logra visualizar por completo el total de los módulos sin retraso alguno. Cumple con los criterios de aceptación planteados.

**Tabla 8**

información de la tarea FLYKE-37.

<b>[FLYKE-37] Crear interfaz de manejo de empleados</b>	
<b>Estado:</b>	Enviada a revisión
<b>Proyecto:</b>	FLYKE
<b>Tipo:</b>	Tarea
<b>Informador:</b>	Sebastián Pachón
<b>Resolución:</b>	Listo
<b>Prioridad:</b>	Medium
<b>Dev Responsable:</b>	Ivan Quintero Torres
<b>QA Responsable:</b>	Miguel Herrera
<b>Sprint:</b>	Tablero Sprint 1.
<b>Story point estimate:</b>	2
<b>Qa Story point estimate:</b>	2
<b>Descripción</b>	<p>Se debe tener una interfaz en donde se puedan agregar/modificar/eliminar empleados y manejar su rol(dinámico: ejemplo: administrador, cajero, mesero). Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>• Se debe poder agregar nombre, rol, PIN de acceso( 4 dígitos).</li> <li>• Se debe poder crear un empleado.</li> <li>• Se debe poder eliminar un empleado.</li> <li>• Se debe poder modificar la información de un empleado.</li> <li>• Se debe poder listar los empleados actuales.</li> </ul>

En esta tarea se busco la inserción de una nueva interfaz en donde se pudiera visualizar la información de los empleados agregados, así como un CRUD que permitiera la administración de estos.

**Ejecución de pruebas para FLYKE-37**

en esta tarea se realizaron pruebas referentes a la usabilidad, así como de interfaces buscando que estas fueran intuitivas para el usuario así como validar su correcto funcionamiento. también se empezó a trabajar el tema de asignación de roles.

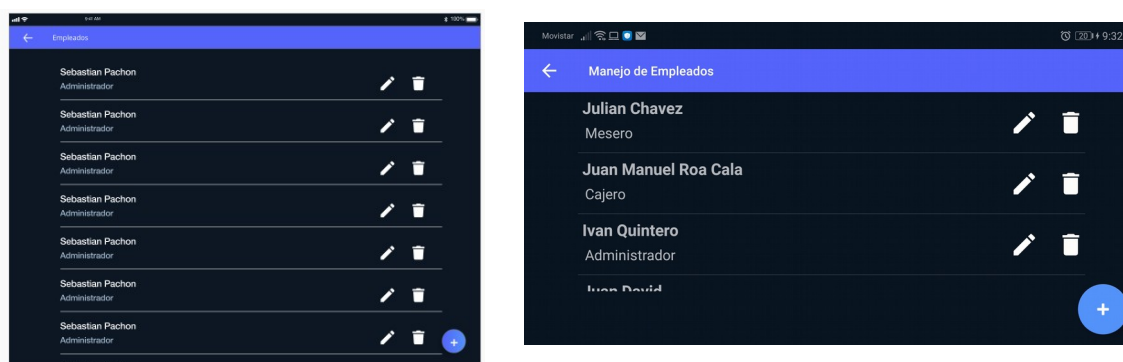
## Tabla 9

### Prueba de interfaz FLYKE-37

Tipo de prueba realizado	Prueba de interfaz
Pasos de la Prueba a realizar	Comparativa entre vistas diseñadas en adobe XD vs las interfaz desarrollada
Resultados Esperados	Las vistas desarrolladas dentro de la actividad corresponden al mismo diseño de las que se planearon en el diseño de interfaces, logrando la ubicación de los diferentes elementos de esta.
Resultados Reales	Las interfaces definidas son aceptadas y cumplen con lo requerido aunque tuvieron un retraso de entrega de la actividad debido a la no entrega de las vistas completas de diseño por parte del product owner. Se hace la observación para evitar este tipo de situaciones además que el pin de cada empleado se puede duplicar lo cual genera inconsistencias con la funcionalidad.

## Figura 10

comparativa vistas diseñada vs vista implementada. FLYKE-37



nota. A la izquierda la vista de empleados creada en la herramienta adobe XD, a la derecha la interfaz implementada en le proyecto.

### ***Retroalimentación de Sprint***

Para este sprint se tuvo en cuenta el uso constante de la herramienta ADOBE XD para la verificación de vistas.

**Semana 1** se realiza la comprobación de herramientas y recursos a manejar para el sprint ( emuladores, diseño de vistas. tablero de actividades)

**semana 2** se empiezan a realizar las primeras pruebas a las tareas de inicio de sesión, y aunque no estuvo finalizada, tuvo un muy buen avance en cuanto a diseño y seguridad.

**Semana 3** se continua con la revisión y se espera la entrega de la actividad FLYKE-37

**semana 4** se realiza validación de los criterios de aceptación de la tarea de gestión de empleados. El pin que se configura a cada empleado se puede repetir lo cual no se considera seguro y se crea una nueva incidencia con el error.

### **7.3 Sprint. 2**

En este sprint se empieza estudiar temas relacionados a pruebas Api-Rest, buscando la generación de consultas a la aplicación a través del servicio consumido, así se puede validar, los tiempos de respuesta al momento de ejecutar dicha consulta, además de validar el nivel inserción de nuevos elementos directamente al aplicativo desde este servicio, además se estudia la creación de pruebas automatizadas utilizando diferentes herramientas, esto con el fin de optimizar tiempos de al momento de hacer pruebas al API, y que estas se generen de manera automática cada vez que se registre un cambio ya sea en su contenido o estructura. También se continúa evaluando las interfaces desarrolladas, se empiezan a ver las primeras funcionalidades del programa.

Tabla 10

*Información de la actividad FLYKE-34*

<b>[FLYKE-34] Crear la interfaz de inicio de sesión</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Ivan Quintero Torres		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 1, Tablero Sprint 2		
<b>Story point estimate:</b>	4		
<b>Qa Story point estimate:</b>	4		
<b>Descripción</b>	<p>Se requiere que se genere la interfaz de inicio de sesión que corresponderá al inicio único de sesión de la aplicación Flyke</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Seguir el diseño de inicio de sesión.</li> <li>● Se debe tener dos campos, uno de correo, y uno de contraseña. Se debe validar que el campo de correo sea de tipo correo. Y el campo de contraseña sea oculto.</li> <li>● Se debe validar en inicio de sesión el largo de la contraseña y que se muestren errores concordantes.</li> <li>● Se debe en registro validar la existencia de los campos requeridos y el largo de la contraseña y que la confirmación sean iguales</li> </ul>		

*Nota.* En la tabla se puede observar el tipo de actividad a realizar así como los involucrados en el cumplimiento de la misma, tiempos de realización estimados, ruta de ubicación según el sprint asignado, los criterios de aceptación que debe cumplir la tarea para ser válida y el nivel de prioridad que posee.

***Ejecución de pruebas para FLYKE-34***

Para la validación de esta actividad se realizaron pruebas de seguridad, verificando que cada campo en el registro de cuenta de usuario ya que en el sprint anterior no se realizó la

verificaron de esta parte del modulo de inicio de sesión, buscando en esta oportunidad que cumpliera con los parámetros establecidos, además de pruebas de interfaz buscando que se cumpliera la visualización conforme al diseño.

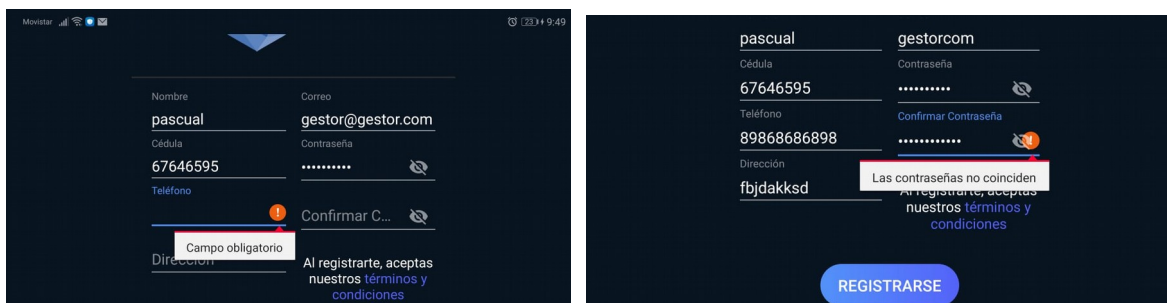
**Tabla 11**

*Prueba de seguridad tarea FLYKE-34*

Tipo de prueba realizado	Prueba de seguridad
Pasos de la Prueba a realizar	Ingresar datos validos y erróneos en cada campo del formulario de registro observar si se valida el tipo de datos ingresados( en correo sea tipo correo, en contraseña tenga ocultamiento de caracteres.) validar que los campos estén completos
Resultados Esperados	Se debe validar correctamente los datos, de ser erróneos que se genere un mensaje de alerta, mostrando al cliente el tipo de error. De la misma manera el formulario de registro de cuenta debe cumplir con parámetros como largo mínimo de caracteres en contraseña y que los datos ingresados sean validados según el tipo de dato.
Resultados Reales	Al ingresar datos que no corresponden al campo de registro estos campos generan el mensaje correspondiente al tipo de error generado, especificando al cliente que esta sucediendo.

**Figura 11**

*Verificación de campos en el formulario de registro.*



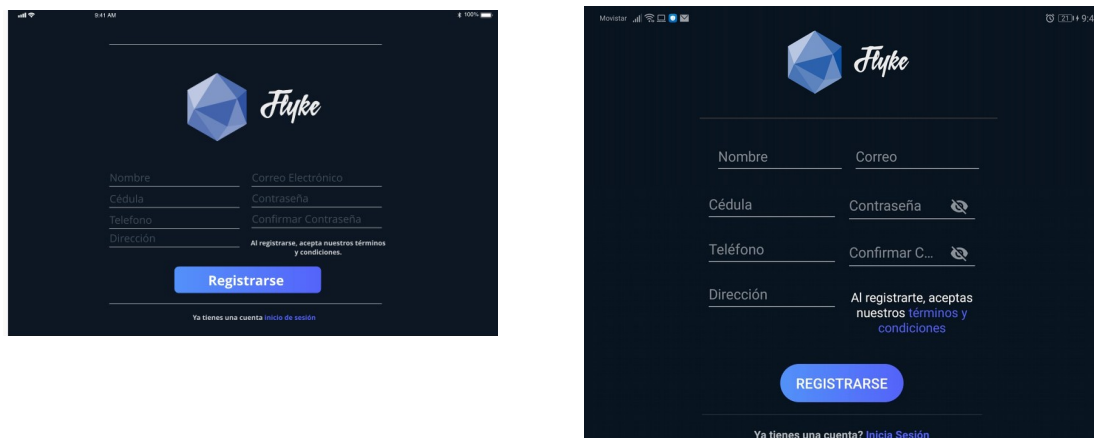
**Tabla 12**

*Prueba de interfaz tarea FLYKE-34*

Tipo de prueba realizado	Prueba de interfaz
Recurso utilizado	Adobe XD
Pasos de prueba a realizar	Comprobación de vistas diseñadas vs interfaces creadas.
Datos de la prueba	Vistas prediseñadas
Resultados esperados	Las interfaces creadas en esta tarea sean lo mas similares posibles a las diseñadas previamente, cumpliendo con la misma posición de botones, campos de texto, tanto la interfaz de registro como la interfaz de inicio de sesión
Resultados reales	La interfaz de registro cumple con los parámetros generales de composición de la interfaz, mostrando los botones, campos de texto en una posición adecuada para el usuario.

**Figura 12**

*Comparativa interfaz de registro.*



*nota.* A la derecha es la interfaz encontrada en adobe XD y al derecha la interfaz implementada en el proyecto.

**Tabla 13**

*Información de la actividad FLYKE-36*

<b>[FLYKE-36] Crear la interfaz de selección de productos</b>	
<b>Estado:</b>	Finalizada



<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		
<b>Resolución:</b>	Listo		
<b>Dev Responsable</b>	Sergio Julian Chavez Chavarro		
<b>Sprint:</b>	Tablero Sprint 2		
<b>Story point estimate:</b>	1		
<b>Qa Story point estimate:</b>	1		
<b>Descripción</b>	<p>Al momento de dar clic en una categoría se debe abrir un listado de productos en una lista normal.          Debe usarse recyclerview con un adaptador.          Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Seguir los diseños acordados.</li> <li>● Mostrar una lista de productos dummy.</li> </ul>		

En esta actividad se buscó visualizar que las interfaces diseñadas tuvieran la mayor solicitud con el diseño preliminar, además que hubiera una correcta continuidad entre las vistas desarrolladas.

### ***Ejecución de pruebas.***

Nuevamente se realizan pruebas de interfaz, así como pruebas de funcionalidad buscando la correcta transición y visualización de cada uno de los elementos.

### **Tabla 14**

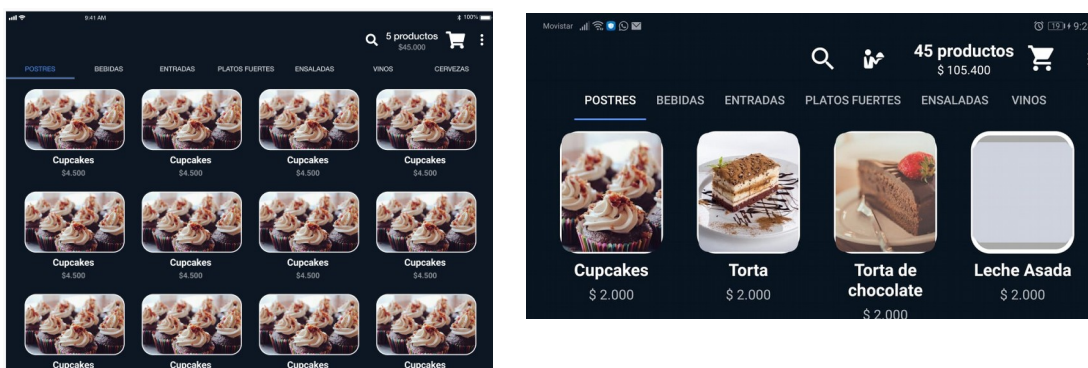
#### *Prueba de interfaz FLYKE-36*

Tipo de prueba realizado	Prueba de interfaz
Pasos de la Prueba a realizar	Comparativa entre vistas diseñadas en adobe XD vs las interfaz desarrollada
Resultados Esperados	Las vistas desarrolladas dentro de la actividad corresponden al mismo diseño de las que se planearon en el diseño de interfaces, logrando la ubicación de los diferentes elementos de esta.
Resultados Reales	Las interfaces son similares y cumplen con el criterio de aceptación básico, se logra identificar cada uno de los elementos presentes,

además no se saturan los colores, el tamaño de los elementos es correcto.

**Figura 13**

*comparativa interfaz diseñada vs interfaz implementada tarea FLYKE-36*



*nota.* a la derecha la interfaz implementada en el proyecto, y a la izquierda la vista diseñada en adobe XD

**Tabla 15**

*Prueba de interfaz FLYKE-36*

Tipo de prueba realizado	Prueba de funcionalidad
Pasos de la Prueba a realizar	Ingresar al aplicativo, ir a modulo general de categorías, seleccionar modulo de facturación. Visualizar una lista dummy de productos.
Resultados Esperados	Al momento de realizar una selección de categoría esta debe mostrar una lista de productos.
Resultados Reales	Se selecciono la categoría en el modulo principal, y esta muestra una lista dummy de productos, clasificados según el tipo del producto

**Tabla 16**

*Información de la actividad FLYKE-41*

**[FLYKE-41] Agregar un producto a un pedido directamente si no tiene ingredientes ni extras**

**Estado:** Finalizada

<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Sebastián Pachón		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 2		
<b>Story point estimate:</b>	6		
<b>Dev Story point estimate:</b>	4		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>En la pantalla de categorías si un producto no tiene ingredientes ni adiciones debe poder agregarse directamente al pedido sin ir a la pantalla de producto.</p> <p><b>Criterios de aceptación:</b></p> <ul style="list-style-type: none"> <li>● Si un producto no tiene ingredientes ni adiciones debe agregarse directamente al pedido activo</li> </ul>		

Esta actividad ya agrega una funcionalidad de selección de productos según si contiene ingredientes adicionales o si es un producto que no permite la edición de mas componentes.

### ***Ejecución de pruebas.***

Dentro del marco de actividades para la comprobación de la funcionalidad se realizo pruebas de usabilidad, respetando los criterios de aceptación.

### **Tabla 17**

#### *Prueba de funcionalidad FLYKE-41*

Tipo de prueba realizado	Prueba de usabilidad.
Pasos de la Prueba a realizar	Ingresar a la aplicativo, ir al modulo principal, ir a modulo de facturación, selección de productos que no incluyan ingredientes adicionales.
Resultados Esperados	El producto que no incluye ingredientes adicionales, se agrega directamente al pedido sin necesidad de abrir la vista de selección de ingredientes.
Resultados Reales	Al momento de seleccionar la categoría de facturación, el sistema abre la lista de productos incluida, al seleccionar cada uno de los

	elementos, los que contienen un ingrediente adicional se abren las opciones en una nueva ventana, y los que no contienen ese tipo de adiciones pasan
--	--

**Tabla 18**

*Información de la actividad FLYKE-40*

<b>[FLYKE-40] Validar ingredientes antes de agregarlo a un pedido</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		
<b>Resolución:</b>	Listo		
<b>Sprint:</b>	Tablero Sprint 2		
<b>Story point estimate:</b>	5		
<b>Dev Story point estimate:</b>	3		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>En la pantalla de producto si el producto tiene ingredientes obligatorios se debe validar que estén seleccionados correctamente antes de poder agregarlo al menú.</p> <p><b>Criterio de aceptación</b></p> <ul style="list-style-type: none"> <li>● Se debe validar que si un producto tiene ingredientes obligatorios no le permite agregar el producto al pedido</li> </ul>		

En estas dos actividades se validó el ingreso de ingredientes al pedido que realiza el cliente. Se tiene en cuenta que las funcionalidades se cumplan, respetando las interfaces de diseño y la continuidad entre vistas.

***Ejecución de pruebas.***

Para este caso se valido que un producto con ingredientes obligatorios no se agregara directamente al pedido si no que le permitiera al usuario la selección de aquellos ingredientes obligatorios.

### Tabla 19

#### *Prueba de funcionalidad FLYKE-40*

Tipo de prueba realizado	Prueba de usabilidad.
Pasos de la Prueba a realizar	Ingresar a la aplicativo, ir al modulo principal, ir a modulo de facturación, selección de productos que incluya ingredientes obligatorios.
Resultados Esperados	El producto que incluya ingredientes adicionales le permita al usuario la selección de dichos subproductos, abriendo una nueva vista para la selección de los mismos.
Resultados Reales	Al momento de seleccionar la categoría de facturación, el sistema abre la lista de productos incluida, al seleccionar cada uno de los elementos, los que contienen un ingrediente adicional se abren las opciones en una nueva ventana, y da la opción de selección de la cantidad de ingredientes y variantes de acompañantes del producto. Estas adiciones se hacen obligatorias ya que hay algunos que son imprescindibles para un pedido ejemplo( para una hamburguesa obligatoriamente debe llevar un tipo de pan)

#### ***Retroalimentación de Sprint***

**Semana 1** se realiza la comprobación de la tarea pendiente desde el sprint anterior referente al inicio de sesión y registro,, verificando que todo el sistemas de login funcione de manera correcta.

**semana 2** se realizan pruebas referentes a la selección de categorías.

**Semana 3 y 4.** se hacen validación de criterios de aceptación de toda la sección referente a inclusión de productos cuando estos tienen ingredientes obligatorio y cuando no. verificando la funcionalidad correcta, la similitudes entre interfaces.

### 7.4 Sprint 3.

Para este sprint se retoman algunas tareas que requieren correcciones o que se les ha generado incidencia, también se implementan nuevas vistas lo que implica una validación constante en referencia con las interfaces diseñadas inicialmente.

Se verifican los criterios de seguridad de la aplicación, también se realizan pruebas de regresión para validar que las tareas que se van integrando no afecten las funcionalidades ya desarrolladas

#### Tabla 20

Información de la actividad FLYKE-42

[FLYKE-42] Implementar Adiciones			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		
<b>Resolución:</b>	Listo		
<b>Etiquetas:</b>	Ninguno		
<b>Dev Responsable:</b>	Juan David Gutiérrez		
<b>Sprint:</b>	Tablero Sprint 2, Tablero Sprint 3		
<b>Story point estimate:</b>	2		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>Se debe agregar las adiciones a llamado de obtener menú y guardarlo en la base de datos. También se debe mostrar y poder seleccionar en la pantalla de producto.</p> <p><b>Criterios de aceptación:</b></p> <ul style="list-style-type: none"> <li>● En la pantalla de productos las adiciones deben verse reflejadas y poder ser seleccionadas.</li> </ul>		
<b>Comentario</b>			
Falta este criterio (“En la pantalla de categorías se deben ver la lista de opciones seleccionadas en la parte de pedido“)			
Comentado por Sebastian Pachón			
El criterio se pasó a otro tiquete, voy a actualizar la descripción			

### Ejecución de pruebas.

Para este caso se validó que las adiciones creadas para el restaurante puedan ser agregadas al pedido para ello se validó a través de pruebas funcionales que se cumpliera todo el proceso.

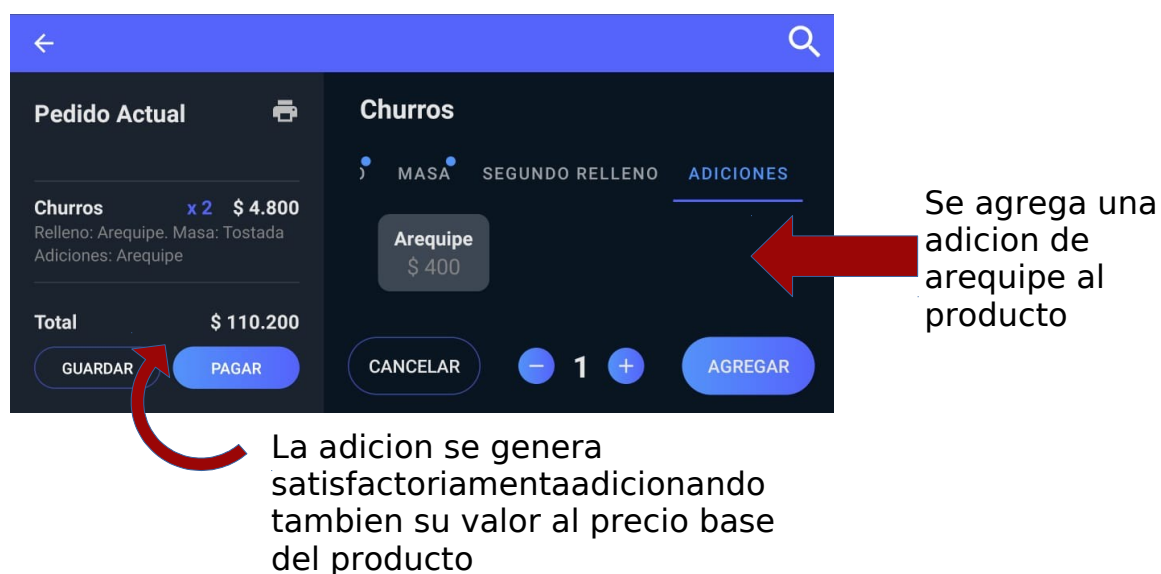
**Tabla 21**

*Prueba de funcionalidad FLYKE-42*

Tipo de prueba realizado	Prueba de usabilidad.
Pasos de la Prueba a realizar	Ingresar adiciones al pedido y que estas se vean reflejadas en la factura.
Resultados Esperados	Al momento de ingresar adiciones al producto estas se deben agregar al pedido sin que genere ningún contratiempo.
Resultados Reales	Las adiciones se agregan satisfactoriamente y son reflejadas en el pedido de manera que se cumple los criterios de aceptación planteados.

**Figura 14**

*agregar adiciones al pedido*



**Tabla 22**

*Información de la actividad FLYKE-43*

<b>[FLYKE-43] Implementar APIs en la vista de empleados</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Ivan Quintero Torres		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 2, Tablero Sprint 3		
<b>Story point estimate:</b>	4		
<b>Qa Story point estimate:</b>	2		
<b>Descripción.</b>	<p>En la vista de empleados se debe utilizar las APIs para la conexión con empleados en vez de utilizar datos quemados.</p> <p><b>Criterios de aceptación:</b></p> <ul style="list-style-type: none"> <li>● Se debe poder visualizar la lista de empleados de un merchant,</li> <li>● Se debe poder agregar un empleado.</li> <li>● Se debe poder remover un empleado.</li> </ul>		

Para esta tarea se utiliza la herramienta postman en donde se hace la creación de empleados a través de esta herramienta, empezando a dar funcionalidad a este módulo.

En donde sea visible la opción de agregar un empleado, poderlo mover y poderlo visualizar en el api.

### ***Ejecución de pruebas.***

Se trabajo a través de postman para la gestión de pruebas de API-REST buscando la respuesta y la inserción de nuevos datos a través del servicio consumido por la aplicación.

### **Tabla 23**

#### *Prueba de funcionalidad FLYKE-43*

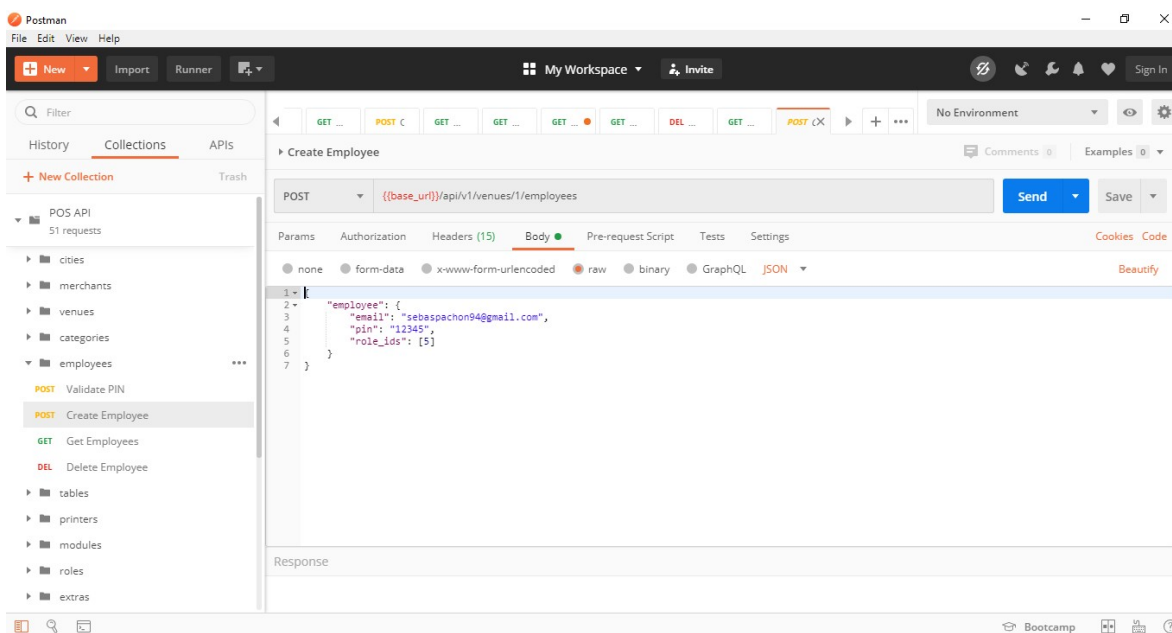
Tipo de prueba realizado	Prueba de api-rest
Pasos de la Prueba a realizar	Poder visualizar un usuario a través del api con el uso de postman.



Resultados Esperados	Visualizar las opciones de insercion, eliminacion y creaci3n de empleados a traves del api
Resultados Reales	En las colecciones del api se puede observar las opciones de get, create y delete, que corresponden a la creaci3n, edici3n y supresi3n de un empleado as3 como la asignaci3n del pin de seguridad asignado a cada.

**Figura 15**

*Interfaz de postman con vista a la creaci3n, y edici3n de empleados.*



En esta etapa se dio un poco mas de indagaci3n a la herramienta postman. En este punto se busc3 la creaci3n de empleados, as3 como la edici3n de estos, a estas alturas los empleados creados hab3an de tener un pin para el acceso, no se tiene asignaci3n de permisos aun.

**Tabla 24**

*Informaci3n de la actividad FLYKE-49*

<b>[FLYKE-49] Agregar funcionalidad de buscador en 3rdenes</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium

<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Juan David Gutiérrez		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 3		
<b>Story point estimate:</b>	5		
<b>Dev Story point estimate:</b>	4		
<b>Qa Story point estimate:</b>	1		
<b>Descripción</b>	<p>Se debe poder filtrar categorías y productos a partir de lo escrito por el usuario. Ejemplo: Existe el producto churro en postres y churrasco en platos fuertes. El resultado el esperado es que solo muestre las categorías de postres y platos fuertes y solo el producto correspondiente Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Debe filtrar en tiempo real a medida que va escribiendo en el texto</li> <li>● Debe seguir el resultado esperado.</li> <li>● En caso de que la consulta arroje cero resultados mostrar un mensaje acorde a esto</li> </ul>		

Para esta actividad se evaluó que se realizará el filtro adecuado en la barra de búsqueda, y que se visualizará según el nombre del producto y categoría al cual corresponde.

### ***Ejecución de pruebas.***

Se realiza prueba de funcionalidad buscando que el buscador filtre según lo digitado en el campo de texto, que se filtre según el producto y según la categoría a la cual pertenece ese producto.

### **Tabla 25**

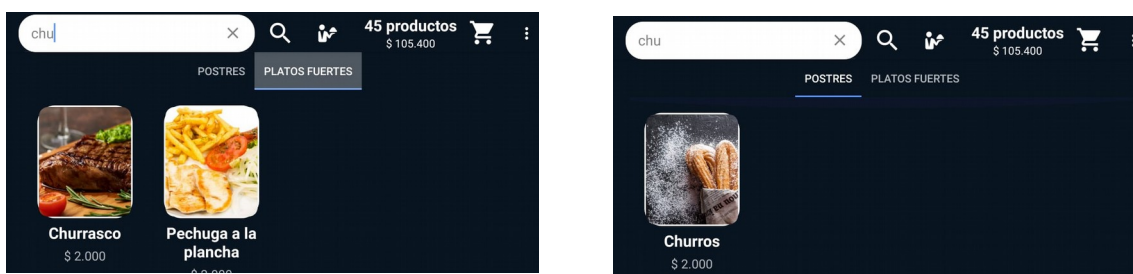
#### *Prueba de funcionalidad FLYKE-49*

Tipo de prueba realizado	Prueba de funcionalidad.
Pasos de la Prueba a realizar	Se debe digitar sobre el buscador de productos y este filtra según lo escrito.
Resultados Esperados	Al momento de escribir el nombre de un producto se debe ir

	filtrando conforme a lo escrito.
Resultados Reales	Se escribieron diferentes palabras buscando la adecuada funcionalidad del buscador de productos, hace un filtro rápido no solo por el producto si no también por la categoría a la cual pertenece este. Ejemplo( se escribe “chu” haciendo referencia a churrasco, el sistema me muestra el churro en su categoría postres y churrasco en su categoría de parrilla.)

**Figura 16**

*implementación del buscador de productos.*



**Tabla 26**

*Información de la actividad FLYKE-54*

<b>[FLYKE-54] al asignar un pin ya creado para otro usuario, no hay mensaje de error.</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Error	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Miguel Herrera	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 3		
<b>Descripción</b>	<p>cuando se asigna un pin a un empleado y este pin ya fue creado a otro empleado la aplicación no genera una respuesta de error y aparece símbolo de cargando.</p> <p>Que debería hacer: al ingresar un pin que ya ha sido asignado, mostrar un mensaje de error de que el pin ya fue asignado.</p> <p>ubicación de bug:</p> <ul style="list-style-type: none"> <li>● iniciar sesión</li> <li>● ingresar pin de acceso</li> </ul>		

- seleccionar módulo empleados
- seleccionar un empleado creado con anterioridad y
- asignar un pin que esté asignado a otro cliente

### ***Ejecución de pruebas.***

Se realizaron pruebas de seguridad buscando que el error referente a la duplicación de pin de los usuarios sea solucionado

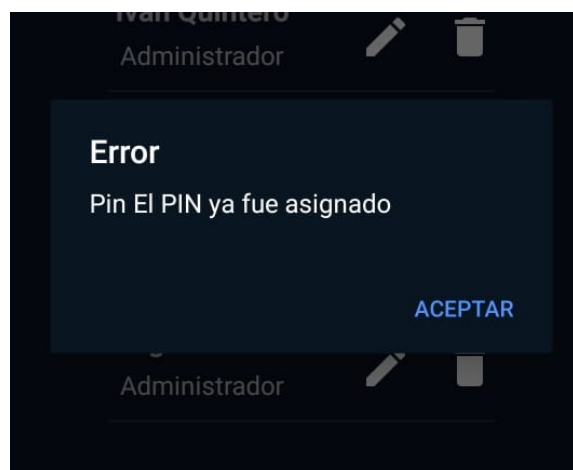
### **Tabla 27**

#### *Prueba de seguridad FLYKE-54*

Tipo de prueba realizado	Prueba de seguridad
Pasos de la Prueba a realizar	Acceder a la configuración de empleado y asignar pines repetidos y observar el comportamiento.
Resultados Esperados	Cada empleado tendrá su pin único el cual no podría ser usado por otro empleado
Resultados Reales	El pin asignado por cada empleado el cual debe ser único corresponde a lo planteado en el requerimiento de la solución. Mostrando un error referente cuando se hace una duplicidad de código pin.

### **Figura 17**

*verificación de mensaje de error pin duplicado.*



Con la identificación de errores se crea una nueva incidencia como en este caso, se encontró a través de pruebas de seguridad que el pin asignado para un empleado se puede asignar a otro usuario, se escribe una descripción del error encontrado y la ubicación del mismo, así como una breve descripción de lo que debería hacer, posteriormente se asigna al responsable de desarrollo para que verifique y dé solución a esta tarea. Para este caso se da solución a esta tarea ubicada en los primeros sprints luego de ser creada una incidencia.

### ***Retroalimentación de Sprint***

**Semana 1** se reviso las actividades referentes a la búsqueda de productos, filtrado por producto y categorías.

**semana 2** se realizaron pruebas al api con el fin de identificar la gestión de empleados a través del servicio consumido por el aplicativo

**Semana 3** se verifico que las adiciones creadas al momento de ser ingresadas al pedido estas se vean reflejadas en el listado de facturación garantizando la realización correcta del pedido y su liquidación

**semana 4** se hizo correcciones exhaustivas al modulo de empleados permitiendo la validación del error referente a la duplicidad de pines de usuarios, lo cual aseguraba un riesgo de seguridad y el cual fue mitigado con satisfacción.

### **7.5 Sprint 4**

En este sprint se incorporan funcionalidades referentes al módulo de facturación en la aplicación FLYKE, se evalúa el comportamiento de la diferentes vistas ya incorporadas y la interacción entre las mismas, así como el funcionamiento del seguimiento de los pedidos realizados según la distribución de las mesas del negocio

En el proceso de aseguramiento de la calidad de las pruebas hechas en las diferentes tareas se van creando incidencias, estas son bugs o errores que surgen y que son dadas a conocer al equipo de desarrollo a través de el tablero de actividades alojado en la aplicación JIRA, la cual ayuda a la planificación de los sprints , dando un tiempo estimado y dando la posibilidad de asignar responsables tanto de corregir dichos bugs como para realizar las pruebas a dicha tarea.

**Tabla 28**

*Información de la actividad FLYKE-44*

<b>[FLYKE-44] Implementar vista de pago de pedido</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Ivan Quintero Torres (Inactivo)		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 3, Tablero Sprint 4		
<b>Story point estimate:</b>	2		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>Crear vista de pago de pedido una vez se hayan seleccionado los productos para un pedido. Se debe poder seleccionar los diferentes tipos de pago disponible y proceder con el pago y el envío al API.</p> <p><b>Criterios de aceptación:</b></p> <ul style="list-style-type: none"> <li>● Se debe tener la facultad de elegir el tipo de pago que se va a realizar (Efectivo, Tarjeta, etc)</li> <li>● Se debe tener un resumen del pedido así como un total.</li> <li>● Se debe enviar el llamado al servidor para guardar el pedido con sus productos y diferentes ingredientes y adiciones seleccionados.</li> </ul>		

**Tabla 29***Información de la actividad FLYKE-52*

<b>[FLYKE-52] Recargar el pedido con la información anterior si un pedido está activo</b>											
<b>Estado:</b>	Finalizada										
<b>Proyecto:</b>	FLYKE										
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium								
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera								
<b>Resolución:</b>	Listo										
<b>Enlaces de incidencias:</b>	<table border="1"> <thead> <tr> <th colspan="4"><b>Relates</b></th> </tr> </thead> <tbody> <tr> <td>relates to</td> <td>FLYKE-50</td> <td>Implementar vista de mesas</td> <td>Finalizada</td> </tr> </tbody> </table>			<b>Relates</b>				relates to	FLYKE-50	Implementar vista de mesas	Finalizada
<b>Relates</b>											
relates to	FLYKE-50	Implementar vista de mesas	Finalizada								
<b>Dev Responsable:</b>	Juan David Gutiérrez (Inactivo)										
<b>Sprint:</b>	Tablero Sprint 3, Tablero Sprint 4										
<b>Story point estimate:</b>	6										
<b>Dev Story point estimate:</b>	5										
<b>Qa Story point estimate:</b>	1										
<b>Descripción</b>	<p>Si un pedido en una mesa se encuentra con el estado activo al momento de abrir la mesa se debe cargar lo que ya fue ordenado con anterioridad y poder agregar elementos nuevos</p> <p>Criterios de aceptación:</p> <ul style="list-style-type: none"> <li>● Si la mesa tiene un pedido activo se debe reflejar lo pedido anteriormente en la pantalla de selección de productos</li> <li>● Se deben poder agregar items nuevos al pedido sin agregar los ya existentes</li> </ul>										

estas tareas entran a ser parte de la creación de las funcionalidades de agregar productos a un pedido y poder visualizarlo y consultarlo después, sin que se reinicie el pedido a cero. Se realizan pruebas de funcionalidad, revisando que cada parámetro se cumpla y que se mantenga de igual manera el diseño de interfaces.

**Tabla 30***Información de la actividad FLYKE-57*

<b>[FLYKE-57] Realizar correcciones a la vista de categorías</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Sebastián Pachón		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 4		
<b>Story point estimate:</b>	8		
<b>Dev Story point estimate:</b>	6		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	<p>Realizar cambios a la interfaz de categorías que será la de toma pedidos:</p> <ul style="list-style-type: none"> <li>● Remover la lista de invoice</li> <li>● Cambiar la vista por una cuadrícula de 4 en vez de 3</li> <li>● Quitar el cardview de los productos</li> <li>● Agregar botón de llamar mesero</li> <li>● Modificar vista de producto como modal (descripción, imagen, nombre, botón de customizar)</li> <li>● El searchview debe estar en el action bar</li> <li>● Debe haber en el actionbar un botón resumen del pedido (total, no productions)</li> </ul>		

en esta tarea se realizó una corrección al diseño propuesta por el cliente el cual buscaba que el aplicativo se acomodará al dispositivo al cual estuvo enfocado (tablet android)

## 7.6 Sprint 5

En estas actividades hay funcionalidades referentes a los métodos de impresión de facturas de pedidos es por eso que se debe tener en cuenta todos los hilos que esto conlleva como los son las impresiones para las diferentes áreas del comercio donde se aplique el sistema,



se siguen incorporando nuevas vistas, a lo cual se le debe realizar minuciosamente pruebas tanto de regresión como de funcionalidad para evitar que los procesos tengan vacíos.

Para esta altura el programa ya tiene algunas funcionalidades de métodos de pagos de productos así como selección de platos para la creación de órdenes y se avanza en el proceso de impresión de dichos pedidos, también se cuenta con el área de gestión de empleados, con sus diferentes roles definidos, lo cual ayuda a definir administradores del sistema y usuarios ocasionales así como sus niveles de acceso, protegiendo datos sensibles y manipulación indebida de los mismos.

### Tabla 31

*Información de la actividad FLYKE-60*

<b>[FLYKE-60] Crear vista de cierre de caja</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Dev Responsable:</b>	Sergio Julian Chavez Chavarro (Inactivo)		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 4, Tablero Sprint 5		
<b>Story point estimate:</b>	3		
<b>Dev Story point estimate:</b>	1		
<b>Qa Story point estimate:</b>	2		
<b>Comentarios</b>	existen pequeñas diferencias en cuanto a tamaño y posición de algunos elementos comparando con la interfaz alojada en adobe XD ( ejemplo: el texto “movimientos de caja”		

Se crean nuevas interfaces constantemente y es necesaria la realización de pruebas de interfaces cada vez que exista un cambio de este tipo. Y debido a la diferenciación entre la

interfaz diseñada y la interfaz diseñada preliminarmente, se hace la observación y se informa al responsable de la tarea para que realice los cambios pertinentes.

**Tabla 32**

*Información de la actividad FLYKE-69*

<b>[FLYKE-69] Corregir cerrar sesión en la vista de módulos</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Estimación original:</b>	Desconocido		
<b>Dev Responsable:</b>	Ivan Quintero Torres		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 5		
<b>Story point estimate:</b>	2		
<b>Dev Story point estimate:</b>	1		
<b>Qa Story point estimate:</b>	1		
<b>Descripción</b>	<p>Actualmente al cerrar sesión cierra la sesión completamente, el botón debería cerrar la sesión de empleado y retornar a la vista de PIN</p> <p><b>Criterios de aceptación:</b></p> <ul style="list-style-type: none"> <li>● Debería cerrar la sesión del empleado, no del administrador.</li> <li>● Debe retornar a la vista de PIN.</li> <li>● Se debe quitar el botón de editar perfil</li> </ul>		

## 7.7 Sprint 6

En este sprint se trabajó en parte con la continuación de revisión de interfaces y funcionalidades, y se trabajó también una actualización de la plataforma de los domicilios que buscaba integrar el método de pago con tarjeta de crédito al aplicativo.

Tabla 33

Información de la actividad FLYKE-77

<b>[FLYKE-77] Regresión Lanzamiento Dely Miguel</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo		
<b>Tiempo Trabajado:</b>	4 days		
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 6		
	<p>Las pruebas realizadas se hicieron mediante una Tablet Lenovo y un celular Huawei, los módulos de restaurante y supermercados funcionan correctamente, los productos se agregan al igual que sus ingredientes sin problema y la validación de los mismos antes de ser agregados. Módulos de Licores, Ropa, Farmacia, Belleza, Detalles no tienen una respuesta del servidor.</p> <p>Al pagar en efectivo el pedido se ejecuta correctamente ya sea en Tablet o en Smartphone.</p> <p>El único fallo es el pago mediante tarjeta con las números de prueba, probé cada una y algunas les faltaba un número al final y aparecía el error, las otras no generaban el cobro.</p> <p>Sugeriría agregar separador de unidades al momento de escribir con cuanto uno va a pagar cuando es en efectivo.</p> <p>segunda sugerencia el botón blanco redondo del carro al agregar un producto no desaparece se ve debajo de Mirar Canasta en el módulo Supermercados.</p>		

En este sprint se trabajó con la plataforma dely domicilios como trabajo secundario, el cual consistió en realizar pruebas de regresión a las actualizaciones que se fueron generando para esta plataforma, concretamente al manejo de funcionalidades de corrección de errores, inclusión de funciones de pago con tarjeta de crédito, para esta tarea fue suministrado las credenciales a un servidor externo el cual permitirá a través de un ambiente controlado, realizar las pruebas pertinentes, cuidando con esto que no afectan la versión que se encuentra en producción.

## 7.8 Sprint 7

Estas actividades corresponden al último sprint en el cual hice parte en el proceso de la pasantía en este se integran vistas de métodos de pago de pedidos, y se realiza pruebas de regresión buscando minimizar los errores ocasionados por dichas integraciones,

**Tabla 34**

*Información de la actividad FLYKE-81*

<b>[FLYKE-81] Crear fragmento de tabs de pago</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón	<b>Responsable:</b>	Miguel Herrera
<b>Resolución:</b>	Listo	<b>Votos:</b>	0
<b>QA Responsable:</b>	Miguel Herrera		
<b>Sprint:</b>	Tablero Sprint 7		
Comentado por Miguel Herrera			
el linearLayout debe subir completamente sin necesidad de deslizar como en el diseño de UI			

Para esta tarea evaluó la funcionalidad de métodos de pago, la división del pago del pedido, así como la corroboración de que las interfaces se acomodan a las planteadas por el cliente, encontrando fallos en algunas interfaces lo cual se anuncia al equipo de desarrollo y se crean las observaciones pertinentes.

**Tabla 35**

*Información de la actividad FLYKE-72*

<b>[FLYKE-72] Ajustar actividad vista de pagos</b>			
<b>Estado:</b>	Finalizada		
<b>Proyecto:</b>	FLYKE		
<b>Tipo:</b>	Tarea	<b>Prioridad:</b>	Medium
<b>Informador:</b>	Sebastián Pachón		

<b>Resolución:</b>	Listo		
<b>Etiquetas:</b>	Ninguno		
<b>Dev Responsable:</b>	Juan David Gutiérrez		
<b>Sprint:</b>	Tablero Sprint 5, Tablero Sprint 6, Tablero Sprint 7		
<b>Story point estimate:</b>	10		
<b>Dev Story point estimate:</b>	8		
<b>Qa Story point estimate:</b>	2		
<b>Descripción</b>	Ajustar la vista de pagos a los nuevos diseños.		

Dentro de todas las fases del proyecto se busco siempre que las interfaces correspondiera a las diseñadas por el *product owner*, además de que los procesos y funcionalidades tuvieran el resultado esperados.

## 8. Definición de pruebas

Dentro de la evolución de cada uno de los sprint se realizaron diferentes pruebas que ayudaron a validar, cada una de las funcionalidades desarrolladas, a continuación se muestra el tipo de pruebas aplicadas, en qué sprint tuvieron participación, los recursos que se utilizaron en cada una de ellas así como el proceso de reporte de incidencias y los procedimientos que se utilizaron para el control de cambios.

**Tabla 36**

*Cronograma de pruebas.*

Tipo de prueba	implicados	Recursos a utilizar	Sprint en el que se realizo									
			0	1	2	3	4	5	6	7		
interfaces de usuario	Tester, product owner, equipo desarrollo	blueStack, appcenter,jira										
usabilidad	Tester, product owner	blueStack, appcenter,jira										
seguridad	Tester, product owner	blueStack, appcenter,jira,postman										
componentes	Tester, product owner,equipo desarrollo	blueStack, appcenter,jira,postman, jenkins										
humo	Tester, product owner	blueStack, appcenter,jira										
Regresión	Tester, product owner	blueStack, appcenter,jira										
Aceptación	Tester, product owner.scrum master	blueStack, appcenter,jira										

s-0. Sprint 0

s-4 Sprint 4

s-1. Sprint 1

s-5. Sprint 5

s-2.Sprint 2

s-6. Sprint 6

s-3.Sprint 3

s-7. Sprint 7

***Formatos para el análisis y reporte de incidencias.***

Para el registro de los resultados de estas pruebas se tiene en cuenta el uso de dos herramientas, en el inicio se tiene de un lado excel, mediante la creación de una tabla donde se describe la incidencia encontrada, ubicación, lo que debería hacer y lo que realmente el error. Posterior a esto se migra el proceso a jira, en donde es más cómodo trabajar y gestionar este proceso.

También se dio uso a la herramienta postman el cual permite verificar la velocidad de respuesta del api consumida así como el tipo de respuesta que genera al incluir diferentes test.

## Figura 18

### Formato de manejo de incidencias en excel

Nombre	Descripcion	Pasos	Resultado esperado	Resultado actual
opcion de editar o eliminar direcciones	el usuario no cuenta con una opcion para editar o eliminar direcciones agregadas, que ya no use o que son obsoletas	<ol style="list-style-type: none"> <li>despues de completar la canasta clic en "continuar"</li> <li>clic en cambiar direccion</li> </ol>	el usuario tenga la opcion de editar direcciones o eliminar aquellas que ya no use	
campo telefonico no validado	el campo de telefono al momento de registro de usuario no se encuentra validado y permite el ingreso de letras y numeros	<ol style="list-style-type: none"> <li>ir a la URL: <a href="https://delystaging.herokuapp.com">https://delystaging.herokuapp.com</a></li> <li>clic en iniciar sesion</li> <li>clic en Registrarse</li> </ol>	que el campo de telefono permita el ingreso de caracteres validos en este caso tipo INT	
		<ol style="list-style-type: none"> <li>despues de tener los pedidos en la canasta, clic en continuar</li> </ol>		

### Control de cambios

Para el control de cambios realizados a las incidencias reportadas siempre se uso jira, si bien es cierto , esta permite el seguimiento de las tareas también permite el reporte al responsable del error, sobre su ubicación, descripción y funcionamiento correcto de la tarea. Este a su vez lo

asignaba a su lista de tareas por hacer, y al momento de dar por corregido el error se enviaba nuevamente para su revisión antes de ser aprobada y aceptada para la integración en el proyecto.

## **9. Resultados**

Dentro de los resultados planteados inicialmente se tuvo que se realizan las pruebas pertinentes que cumplieran con los parámetros y requerimientos exigidos por el cliente, quien solicitaba apoyo para el acompañamiento y seguimiento de pruebas y aseguramiento de calidad en el proceso de los proyectos con los que se el pasante deberá involucrarse y cumplir a cabalidad con las tareas planteadas.

Es por esto que al cliente se le hace una retrospectiva del proceso realizado, así como la entrega de algunos documentos creados, los cuales contienen información referente a los procesos de pruebas, dando a conocer lo implementado hasta ese momento, es por ello que el día 7 de mayo de 2020 se realiza la entrega del trabajo realizado desde el mes de septiembre de 2019 en la empresa y se firma el acta de finalización en común acuerdo entre las partes y se da a conocer el resultado final de la realización de las diferentes pruebas y seguimientos hechos en el proyecto.

Se realiza el documento firmado por el gerente de la compañía en relación a la satisfacción por el trabajo realizado por el pasante

### **Figura 19**

*Certificación de satisfacción del cliente*





07-05-2020

**EL GERENTE DE LA EMPRESA ICE COMPANY S.A.S EDWARD HERNANDEZ MOLANO  
DEL MUNICIPIO DE FUSAGASUGÁ CUNDINAMARCA**

**En cumplimiento de lo establecido en el Acuerdo de Voluntades N° C- 20180920-062  
celebrado entre Ice Company S.A.S y la Universidad de Cundinamarca.**

**CERTIFICA:**

Que el pasante **LUIS MIGUEL HERRERA MONROY**, identificado con cedula de Ciudadanía N°. 1.069.747.867, de Fusagasugá, estudiante de ultimo semestre de la carrera de Ingeniería de Sistemas en la Universidad de Cundinamarca de Fusagasugá cumplió a cabalidad y de manera honorifica las actividades propuestas dentro del marco del proyecto de pasantía **IMPLEMENTACIÓN Y ACOMPAÑAMIENTO DE PROCESOS DE CALIDAD Y TESTING EN EL DESARROLLO DE SOFTWARE A TRAVÉS DE METODOLOGÍA SCRUM PARA LA EMPRESA ICE COMPANY S.A.S**

Cordialmente

  
\_\_\_\_\_  
**EDWARD HERNANDEZ MOLANO**

Gerente general – ingeniero de sistemas  
M.P 25255-331762 CND COPNIA  
Tel:871-9943  
Cel:323 4866540

Calle 25a #2ª – 28 Barrio la Macarena Fusagasugá (Cundinamarca)  
Teléfonos: (091)871-9943 Servicio Urgencias - Celular: 3132694664 Administrativo  
info@icecompany.com.co

## 10. Conclusiones

A través del proyecto realizado se hizo un seguimiento y acompañamiento adecuado, acorde con los requerimientos que poseía el cliente, dando solución a estos , graves del manejo de la metodología SCRUM la cual permitió hacer entregables de funcionalidades que fueron incrementando un producto que al momento de ser entregado cumple con algunas funciones básicas.

Se realizo una correcta documentación de las incidencias observadas dando las pautas para ser ubicadas por el responsable del desarrollo de la tarea que se identifico con fallos encontrados, permitiendo que se mantuviera la calidad en el proceso de solución de dichas incidencias.

Se dio un correcto uso a las herramientas ofrecidas por la empresa para el desarrollo del proyecto, lo que permitió que existiera un buen flujo de información y comunicación entre los interesados en los procesos de asignación de tareas, verificación de criterios, gestión de tiempos de entrega, documentaron de las incidencias, entre otras cosas.

finalmente, teniendo el cuenta el alcance que puede legar a tener este proyecto con su realización total y puesta en marcha, se pueden suplir muchas necesidades referentes a la gestión de pedidos, gestión de empleados y procesos de facturación.

## 11.Referencias

Consultora de tecnología SDOS. (30 de abril de 2018). *¿Qué papel tiene QA en las metodologías ágiles?*. <https://sdos.es/blog/que-papel-tiene-qa-en-las-metodologias-agiles>.

Plataforma de estudios OpenWebinars.(27 de junio de 2019) *Tipos de pruebas de software*.  
<https://openwebinars.net/blog/tipos-de-pruebas-de-software/>

Los Andes Training(23 de agosto de 2017). *Qué son las pruebas de aceptación?*.  
<https://losandestraining.com/2017/08/23/que-son-las-pruebas-de-aceptacion/>

Sinanaps(2020) *SCRUM DASHBOARD: REAL-TIME WORK PROGRESS TRACKING FOR IMPROVED PROJECT PERFORMANCE*. <https://www.sinnaps.com/en/project-management-blog/scrum-dashboard>

Scrum-QA(23 de septiembre de 2010). *Definición Criterio "LISTO"*. <http://scrum-qa.blogspot.com/2010/09/definicion-criterio-listo.html>.

## 12.Tablas

<b>Tabla 1.</b> Cronograma de Actividades	.26
<b>Tabla 2.</b> Información de la actividad FLYKE-34	.29
<b>Tabla 3.</b> Prueba de seguridad tarea FLYKE-34	.31
<b>Tabla 4</b> Prueba de interfaz tarea FLYKE-34	.31
<b>Tabla 5.</b> Información de la actividad FLYKE-35	.32
<b>Tabla 6</b> Prueba de interfaz FLYKE-35	33
<b>Tabla 7</b> Prueba de usabilidad FLYKE-35	.34
<b>Tabla 8</b> información de la tarea FLYKE-37.	.35
<b>Tabla 9</b> Prueba de interfaz FLYKE-37	.36
<b>Tabla 10</b> Información de la actividad FLYKE-34	.38
<b>Tabla 11</b> Prueba de seguridad tarea FLYKE-3	.39
<b>Tabla 12</b> Prueba de interfaz tarea FLYKE-3	.39
<b>Tabla 13</b> Información de la actividad FLYKE-36	.40
<b>Tabla 14</b> Prueba de interfaz FLYKE-36	.41
<b>Tabla 15</b> Prueba de interfaz FLYKE-36	42
<b>Tabla 16</b> Información de la actividad FLYKE-41	.42
<b>Tabla 17</b> Prueba de funcionalidad FLYKE-41	.43
<b>Tabla 18</b> Información de la actividad FLYKE-40	.44
<b>Tabla 19</b> Prueba de funcionalidad FLYKE-40	.45
<b>Tabla 20</b> Información de la actividad FLYKE-42	46
<b>Tabla 21</b> Prueba de funcionalidad FLYKE-42	47
<b>Tabla 22</b> Información de la actividad FLYKE-43	48
<b>Tabla 23</b> Prueba de funcionalidad FLYKE-43	48
<b>Tabla 24</b> Información de la actividad FLYKE-49	49
<b>Tabla 25</b> Prueba de funcionalidad FLYKE-49	50
<b>Tabla 26</b> Información de la actividad FLYKE-54	51
<b>Tabla 27</b> Prueba de seguridad FLYKE-54	52
<b>Tabla 28</b> Información de la actividad FLYKE-44	54
<b>Tabla 29</b> Información de la actividad FLYKE-52	55
<b>Tabla 30</b> Información de la actividad FLYKE-57	56
<b>Tabla 31</b> Información de la actividad FLYKE-60	57
<b>Tabla 32</b> Información de la actividad FLYKE-6	58
<b>Tabla 33</b> Información de la actividad FLYKE-77	59
<b>Tabla 34</b> Información de la actividad FLYKE-81	60
<b>Tabla 35</b> Información de la actividad FLYKE-72	60
<b>Tabla 36</b> Cronograma de pruebas.	62

### 13.Figuras

<b>Figura 1</b> Diferenciación entre error, defecto y fallo.	11
<b>Figura 2</b> Backlog del proyecto FLYKE	20
<b>Figura 3</b> Tablero de actividades de uno de los sprint visualizado en JIRA	21
<b>Figura 4</b> Criterios de aceptación de una tarea en el proyecto FLYKE	23
<b>Figura 5</b> Uso de la herramienta Slack para el control del daily planning y notificaciones de cambios en el proyecto.	24
<b>Figura 6</b> Proceso realizado en cada uno de los sprint	25
<b>Figura 7</b> Verificación de campos en el inicio de sesión	31
<b>Figura 8</b> comparativa interfaz preliminar vs interfaz desarrollada	32
<b>Figura 9</b> vista de categorías. Interfaz diseñada vs interfaz implementada.	34
<b>Figura 10</b> comparativa vistas diseñada vs vista implementada. FLYKE-37	36
<b>Figura 11</b> Verificación de campos en el formulario de registro.	39
<b>Figura 12</b> Comparativa interfaz de registro	40
<b>Figura 13</b> comparativa interfaz diseñada vs interfaz implementada tarea FLYKE-36	42
<b>Figura 14</b> agregar adiciones al pedido	48
<b>Figura 15</b> Interfaz de postman con vista a la creación, y edición de empleados.	49
<b>Figura 16</b> implementación del buscador de productos.	51
<b>Figura 17</b> verificación de mensaje de error pin duplicado	52
<b>Figura 18</b> Formato de manejo de incidencias en excel	53
<b>Figura 19</b> Certificación de satisfacción del cliente	64

### 14.Apéndice

A continuación se presenta alguna documentación que se realizó para el proceso de la pasantía as

## Apéndice A. plan de pruebas

### Introducción

#### 1.1 Propósito

Este documento tiene como propósito establecer las técnicas, herramientas y actividades relacionadas con la ejecución y validación del plan de pruebas; incluye responsabilidades de cada una de las tareas, los recursos y los pre-requisitos que deben ser considerados en el esfuerzo de las pruebas, en la búsqueda de garantizar el cumplimiento de los requerimientos planteados en el marco del desarrollo del proyecto FLYKE

El documento específicamente busca:

- Identifica información del proyecto y componentes de software que pueden ser probados
- Los tipos de pruebas a ser implementados
- Recomendaciones y descripciones de estrategias de prueba a ser empleadas.

#### 1.2 Alcance

Este documento de PLAN DE PRUEBAS, se convierte en una guía para desarrollar de una forma organizada las diferentes actividades que se realizarán en el proceso del plan de pruebas en el desarrollo del proyecto

La metodología scrum aplicada al proyecto principal y este documento de plan de pruebas permitirán al equipo evaluar aspectos como: la funcionalidad, la seguridad, la usabilidad, y realizar un seguimiento de las pruebas realizadas, así como el control de los resultados.

#### 1.3 Definiciones

**El plan de prueba:** describe todos los métodos que se utilizarán para verificar que el software satisface la especificación del producto y las necesidades del cliente. Incluye los objetivos de calidad, necesidades de recursos, cronograma, asignaciones, métodos, etc

**Reporte de pruebas:** describen los problemas encontrados al ejecutar los casos de prueba.

**Herramientas de pruebas y automatización:** documentación de las herramientas empleadas en el proceso de pruebas.

**FLYKE:** nombre del sistema a desarrollar

**Tabla 21**

*Definiciones de pruebas*

Tipo de Prueba	Definición	Fase del proceso
Integración	Permite verificar el correcto ensamblaje entre los distintos módulos que componen el sistema desarrollado, con la creación de nuevas funcionalidades	Desarrollo
Sistema: Interfaz de usuario, seguridad,desempeño	Sistema: Estas pruebas buscan diferencias entre la solución desarrollada y los requerimientos, con el fin de identificar errores que se puedan generar entre la especificación funcional y el diseño del sistema.  Interfaz de usuario: Permite verificar que la navegación a través de los elementos que se están probando, reflejen las funciones del negocio y los requerimientos funcionales.  Seguridad: Verifica el cumplimiento de las	desarrollo

	políticas de seguridad acordadas para el sistema.	
Funcionales	Usabilidad: Esta prueba permite encontrar problemas de factores humanos, en relación a la facilidad de uso, la navegación, diseño amigable, aplicando un conjunto de reglas básicas de usabilidad.	En las iteraciones necesarias
Aceptación	Es la prueba final basada en las especificaciones del usuario o basada en el uso del programa por el usuario final luego de un periodo de tiempo	En la entrega de resultados

## 2.Elementos

### Software sujeto a pruebas

Se evaluará las funcionalidades desarrolladas para el naciente sistema POS llama FLYKE

Se verificará la correcta funcionalidad de las actualizaciones generadas al aplicativo

DELY DOMICILIOS

## 3.Estrategias de Pruebas

### 3.1 Planificación

Se planificaran las pruebas que serán aplicadas específicamente, por lo cual se incluye la definición de cada una de las pruebas, las estrategias , los recursos y las estimaciones de tiempo.

- o Se definen la pruebas a aplicar
- o Se establecen las técnicas a utilizar
- o Se establece un tiempo de ejecución de las pruebas estimado, basado en el tiempo planteado en el sprint planning.
- o Uso de herramientas.



- o Criterios de aceptación para cada requerimiento generado.
- o Recursos involucrados.

### **Resultado de la planificación**

Se deberán definir los siguientes puntos:

- Cronograma de la ejecución de las pruebas; donde se especifica qué prueba se realiza, recursos a utilizar (humanos y tecnológicos).
- Formatos a utilizar para el registro y análisis de los resultados de las pruebas.
- Herramientas a utilizar para la gestión de incidencias.
- Procedimientos para el control de cambios.

### **3.2 Diseño de las Pruebas**

**Definición de casos de pruebas:** se realiza definición, asignación y se prioriza la actividad, además se establece un orden de trabajo, se calcula un tiempo estimado para la comprobación de funcionalidades.

### **3.3. Tipos de Pruebas**

#### **3.3.1. Pruebas de integración**

En esta prueba se comprueba la compatibilidad y funcionalidad de los interfaces entre las distintas ‘partes’ que componen el desarrollo de la solución, es decir esta prueba válida la integración entre los diferentes módulos que componen la solución con el fin de garantizar que su operación integrada es correcta, teniendo en cuenta los siguientes temas técnicos:

- El funcionamiento integrado de módulos debe estar libre de errores.
- Probar todas las dependencias entre módulos

#### **3.3.2. Pruebas funcionales**

el objetivo de las pruebas funcionales es asegurar el trabajo apropiado de los requisitos funcionales tomando como objetos de prueba cada uno de los requerimientos, incluyendo la navegación, entradas de datos, edición, procesamiento y obtención de resultados.

La estrategia de validación que se busca es el uso de set de pruebas definidos, teniendo en cuenta un flujo normal y flujos alternativos de acceso, usando datos válidos e inválidos con el fin de verificar si los resultados esperados ocurren con la inserción de datos verídicos, el despliegue de mensajes de error se hace correctamente cuando se genera datos inválidos.

Como herramientas requeridas para el reporte de incidencias se contará con el uso de la plataforma Atlassian en su módulo jira, el cual permitirá la validación, generación de incidencias y asignación de responsables.

### **3.3.3 . Pruebas de Interfaz de Usuario**

La Prueba de interfaz del Usuario (UI) verifica la interacción de un usuario con el software. Además, estas pruebas deben asegurar, que los objetos dentro de la interfaz funcionan según lo esperado y se conforma con los estándares relacionados a la administración de la salud y a las reglas definidas de Usabilidad

Se tendrá de la misma manera que la prueba anterior el uso de la plataforma atlassian, para el reporte de incidencias y fallos en los casos en los que no se cumplan con los criterios de aceptación planteados.

Las incidencias generadas en este proceso serán atendidas por los desarrolladores responsables de la tarea y se creará una nueva tarea para la corrección de errores.

### **3.3.4 Pruebas de Seguridad**

La prueba de seguridad y de control de acceso se enfoca en dos áreas clave de seguridad:

- Seguridad a nivel aplicación, incluyendo el acceso a los datos o a las funcionalidades del

negocio • Seguridad a nivel sistema, incluyendo el registro en el acceso remoto al sistema. La seguridad a nivel aplicación se asegura de que, basada en la seguridad deseada, restringe a los actores de las funciones específicas o casos de uso, o los limita en los datos que están disponibles para ellos. Por ejemplo, para cada uno se puede permitir incorporar datos y para crear nuevas cuentas, pero solamente los administradores pueden borrarlos.

Estas pruebas tiene como objetivo que se cumpla con el acceso al aplicativo según el nivel de permisos que posee el usuario. Para ellos se verificará cada tipo de usuario, sus credenciales y permisos.

Se dará como criterio de aceptación que los roles actúen conforme al nivel de permisos otorgado.

#### **4.Herramientas para las Pruebas**

Como herramientas para la documentación de las pruebas realizadas se utilizará el aplicativo web atlassian el cual permite seleccionar las tareas de acuerdo al sprint al cual fue asignado, así como la asignación de roles y tiempos estimados de entrega,

Se debe contar además con un emulador de dispositivo android para tablets ya que las funcionalidades desarrolladas deben contar con características de hardware especiales, se enfocara el sistema a este tipo de dispositivos debido a su fácil manejo y portabilidad. Para este caso se usará el emulador BlueStack

Elementos como Postman para evaluar la respuesta de la aplicación por medio del API serán requeridos, esta herramient permite el manejo de solicitudes para visualizar la respuesta del aplicativo a desarrollar.

#### **Software**

para el desarrollo y documentación de las diferentes pruebas a realizar durante el proceso de la pasantía se usará las siguientes herramientas

**Tabla 22**

*Herramientas a utilizar*

<b>nombre</b>	<b>versión</b>	<b>Para que se usa</b>
BlueStacks	Version 4.170.0.1042	Emulador de dispositivos android en plataformas como windows o mac
Jira	Jira software cloud	Clasificador de tareas, organización de tablero de actividades y asignación de roles.
Postman	Version 7.25.0	permite el envío de peticiones HTTP REST sin necesidad de desarrollar un cliente
AppCenter	Versión 2.1.1	Utilizada para ver, descargar y probar las aplicaciones que sufren de versionamiento constante para el tema de realización de pruebas.

### **5.Criterios de Paso/Fallo**

Dentro del proceso de aceptación o rechazo de los resultados de las pruebas realizadas se toma en cuenta la herramienta de planificación de tareas en la cual se listan las tareas a realizar,

Al no cumplirse con los criterios de aceptación de cada tarea, se creará a través del tablero de actividades una incidencia reportando el error al desarrollador encargado.

Debe cumplir con las especificaciones de software planteadas no solo en la funcionalidad sino también en el diseño de la interfaz.

A tomarse como aceptada la tarea se enviará a la sección del tablero de finalización, indicando el tiempo requerido para su evaluación, de ser posible agregar comentarios que orienten al desarrollador.

**Tabla 23**

### *Crterios de paso/fallo*

Actividad		Hitos
Revisiones de Pruebas		
Reuniones de revisión: informal; guiada; técnica; inspección		Reporte de ajuste a realizar
Ajustes a la base de pruebas		Base de Prueba corregida
Pruebas de Integración		
Análisis y diseño de pruebas		Criterios de aceptación
Ejecución de las pruebas		Reporte de
Evaluación de los criterios de salida		Defectos/Incidencias
		Evidencias de Pruebas
Generación de informes y Cierre de Pruebas		Informe de Resultados
		Lecciones Aprendidas
		Productos de prueba
Pruebas de Sistema		
Análisis y diseño de pruebas		Criterios de Aceptación
Ejecución de las pruebas		Reporte de
Evaluación de los criterios de salida		Defectos/Incidencias
		Evidencias de Pruebas
Generación de informes y Cierre de Pruebas		Informe de Resultados
		Lecciones Aprendidas
		Productos de prueba
Pruebas de Aceptación de Usuarios - UAT		
Análisis y diseño de pruebas		Criterios de Aceptación
Ejecución de las pruebas		Reporte de
Evaluación de los criterios de salida		Defectos/Incidencias
Generación de informes y Cierre de Pruebas		Informe de Resultados
		(conformidad)
		Lecciones Aprendidas
		Productos de prueba

## **6.Riesgos**

A continuación se mencionan los posibles riesgos así como los planes de contingencia que se asumen para cada uno.

**Tabla 24**

*Riesgos*

Riesgo	Impacto	Contingencia
Falta de recursos y baja competencia de pruebas	medio	Definir previamente las pruebas, los criterios de entrada y salida.
Falta de recursos necesarios para la ejecución de las pruebas según el plan	alto	Preparación de las pruebas, para garantizar las herramientas, entornos etc.
Tiempo reducido asignado a la fase de pruebas	medio	Ampliación de plazos de entrega, gestionando a través del jefe de proyecto.
Falta de coordinación entre los equipos de desarrollo y testing	medio	Inmersión previa al inicio del proyecto buscando la integración de todos los involucrados
Falta de experiencia con nuevas tecnologías, herramientas y lenguajes de programación.	alto	Indagación preliminar sobre, los procesos a realizar, así como metodologías empleadas.

**Apéndice B. Informe de Sprint**



**INFORME**

**SPRINT 1**

**LUIS MIGUEL HERRERA MONROY**

[mherrera@icecompany.com.co](mailto:mherrera@icecompany.com.co)

Ice company sas- Proyecto POS Flyke Sprint 1		
<b>Tarea:</b> Crear la interfaz de selección de categoría		
<b>Código:</b> FLYKE-35	<b>Responsable:</b> Juan David Gutiérrez	
<b>Descripción:</b> Al iniciar sesión se debe redirigir a la interfaz de selección de categoría. Debe salir un listado cuadrulado de las categorías disponibles. Debe hacerse con la ayuda un recyclerview y un adaptador. (Dinámica)		
Crterios de Aceptación	Validación	Observaciones
Se debe llegar después de hacer inicio de sesión.	!	Se validó al iniciar sesión con la cuenta <a href="mailto:sebaspachon94@gmail.com">sebaspachon94@gmail.com</a> Después de iniciar sesión se redirige a los módulos de la Aplicación. No a la sección de categorías.
Se debe mostrar una lista de categorías dummy, que permitan visualizarse como una cuadrícula.	✓	
Al dar clic en una categoría debe redirigirlo a la interfaz de productos de una categoría	✓	
<b>Observaciones generales:</b> Se debe completar el diseño de la interfaz ya que la barra de Búsqueda no está en funcionamiento.		

\*se utilizó la herramienta “BlueStack versión 4.150.11.1001” para la simulación de Android en Windows

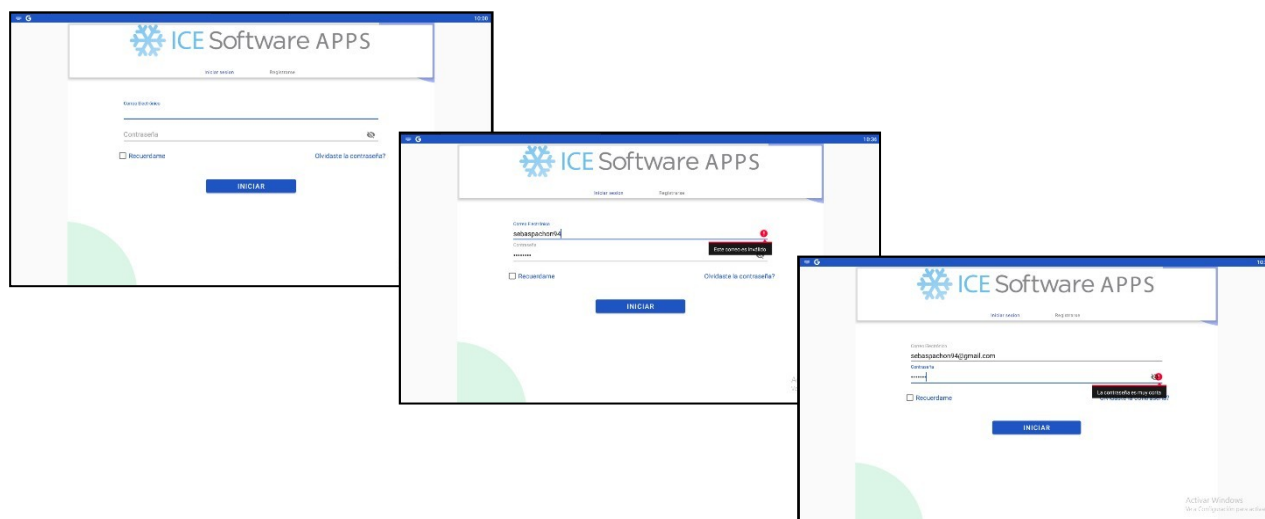




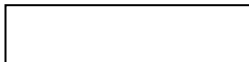
Ice company sas- Proyecto POS Flyke Sprint 1		
<b>Tarea:</b> Crear la interfaz de inicio de sesión		
<b>Código:</b> FLYKE-34	<b>Responsable:</b> Iván Quintero Torres	
<b>Descripción:</b> Se requiere que se genere la interfaz de inicio de sesión que corresponderá al inicio único de sesión de la aplicación Flyke.		
Crterios de Aceptación	Validación	Observaciones
Seguir el diseño de inicio de sesión.	✓	
Se debe tener dos campos, uno de correo, y uno de contraseña. Se debe validar que el campo de correo sea de tipo correo. Y el campo de contraseña sea oculto.	✓	
Se debe validar en inicio de sesión el largo de la contraseña y que se muestran errores concordantes.	✓	
Se debe en registro validar la existencia de los campos requeridos y el largo de la contraseña y que la confirmación sea igual	✓	
<b>Observaciones generales:</b>		

\*se utilizó la herramienta “BlueStack versión 4.150.11.1001” para la simulación de Android en Windows

### Inicio de



# Registro



ICE Software APPS

Inicio Inicio Registro de

Nombre:

Correo:  Correo obligatorio

Celular:  Confirmar

Telefono:  Confirmar Direccion

Direccion:

Al registrarse, aceptas nuestros terminos y condiciones

**REGISTRARSE**

Activar Windows  
Ve a configuración para activar

ICE Software APPS

Inicio Inicio Registro de

Nombre:  Correo

Celular:  1225682745

Telefono:  3723564892

Direccion:  correo@icecompany.com.co

Al registrarse, aceptas nuestros terminos y condiciones

**REGISTRARSE**

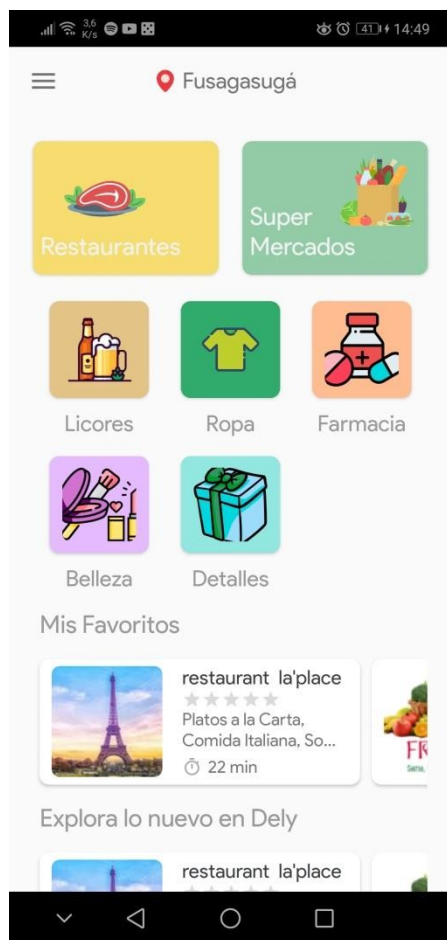
Activar Windows  
Ve a configuración para activar

**Apéndice C.Pruebas de regresión**

Informe prueba de regresión en app móvil  
dely domicilios

Luis miguel herrera Monroy  
[mherrera@icecompany.com](mailto:mherrera@icecompany.com)

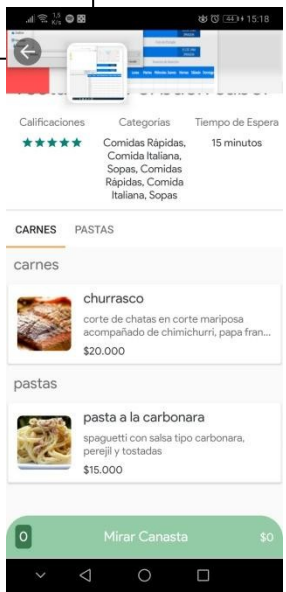
Los modelos de negocio recientemente agregados ( licores, ropa, farmacia, belleza, detalles) aún no están en funcionamiento y producen un error de conexión con el servidor



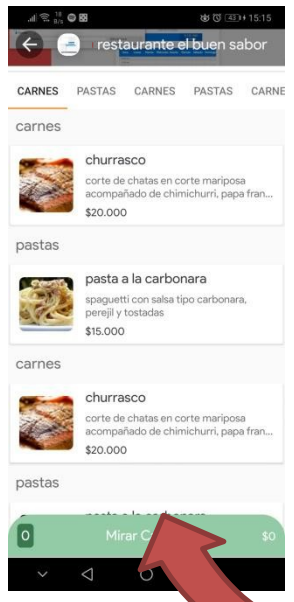
Al actualizar la lista de productos de un restaurante, esta lista se duplica, es decir se duplican los mismos productos y por ende las categorías a la cual corresponden y al seguir actualizando una y

otra vez se agrega un lote de productos más. Sucede con todos los restaurantes sin importar la ciudad.

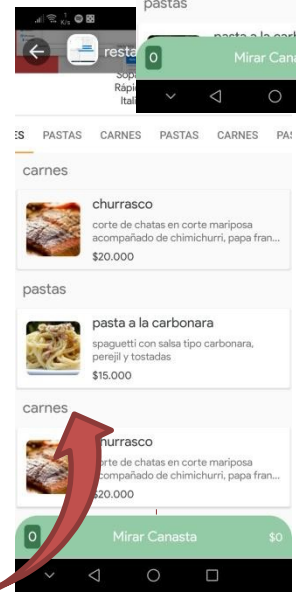
Las dos únicas categorías de productos. Carnes y pastas



Después de actualizar la lista de productos aparecen mas categorías y productos ( duplicados)



Después de actualizar varias veces, se observan varios duplicados de categorías y productos



**Apéndice C. pruebas a interfaz web**

Informe de pruebas realizadas a la interfaz  
web del panel de administración de dely  
domicilios

Luis miguel herrera Monroy  
[mherrera@icecompany.com](mailto:mherrera@icecompany.com)

Error: el restaurante no puede visualizar los detalles del pedido en el módulo de billetera. El botón de “mirar pedido” no funciona

Ruta: <https://delystaging.herokuapp.com/radmins/billetera>



---

Error: no hay manera de validar la distancia del domicilio cuando es subido por el módulo “arma pedidos” por ende el cobro será el mínimo del domicilio, sin importar que sea al otro lado de la ciudad.

Ruta:

[https://delystaging.herokuapp.com/radmins/new\\_order/supermercado-fusafruver/summary](https://delystaging.herokuapp.com/radmins/new_order/supermercado-fusafruver/summary)

**Resumen del Pedido**

maiz	\$2.000	-	1	+
guayaba	\$1.500	-	1	+

**Datos del Cliente**

Nombre del Cliente...

Dirección del Cliente...

Número de Teléfono...

Agrega un comentario para el pedido

Realizar Pedido

Error: el módulo de “arma pedidos” no tiene la opción de tiempo en la que el domiciliario debe pasar a recoger el pedido y automáticamente coloca 20 minutos. Ruta: :

[https://delystaging.herokuapp.com/radmins/new\\_order/supermercado-fusafruver/summary](https://delystaging.herokuapp.com/radmins/new_order/supermercado-fusafruver/summary)

**Resumen del Pedido**

maiz	\$2.000	-	1	+
guayaba	\$1.500	-	1	+

**Datos del Cliente**

Nombre del Cliente...

Dirección del Cliente...

Número de Teléfono...

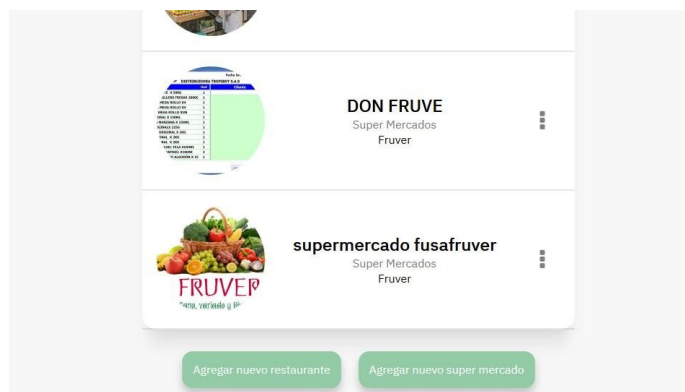
Agrega un comentario para el pedido

Realizar Pedido



Error: no es posible agregar nuevos negocios a las categorías que se han creado (licores, farmacia, detalles, belleza) no existe la opción de asignarle ese tipo de categorías a los nuevos negocios.

Ruta: <https://delystaging.herokuapp.com/admin/restaurantes>



Error: en la descripción de gráficos del módulo de estadísticas, esta no está totalmente optimizada para el reconocimiento de caracteres especiales y de acentuación.

Ruta: <https://delystaging.herokuapp.com/admin/estadisticas>



Error: no es posible usar el módulo de reportes, se estima que aún está en desarrollo, ya que no genera ninguna información.

Ruta: <https://delystaging.herokuapp.com/reportes/index>



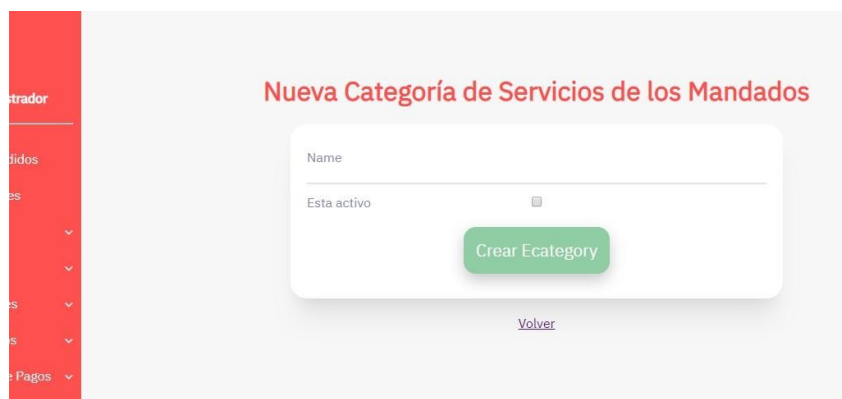
Error: en tipos de negocio los dos últimos elementos de la lista, al abrir las opciones de editar o eliminar salen incompletas o no visibles y son solapadas. Evitando poder acceder a estos submenús, solo con el scroll es posible ver los submenús completos pero no es muy estable ya que al mover un poco el mouse nuevamente se desliza hacia arriba la lista

Ruta: [https://delystaging.herokuapp.com/business\\_types](https://delystaging.herokuapp.com/business_types)



Error: error de escritura en la palabra categoría

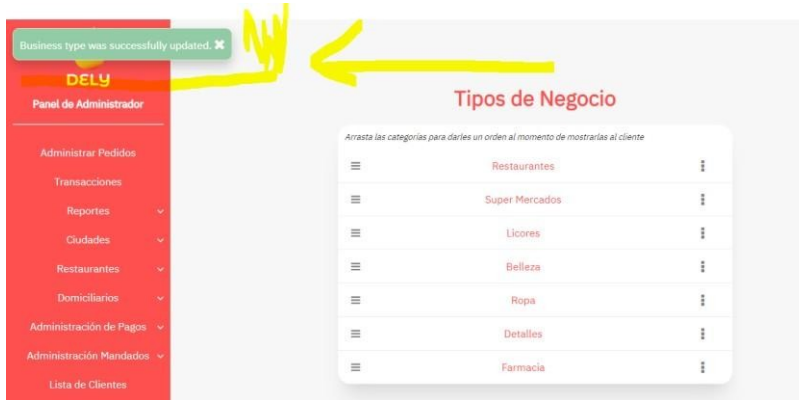
Ruta: <https://delystaging.herokuapp.com/ecategories/new>



Error: al crear un nuevo tipo de negocio o editar uno ya existente, el

mensaje de confirmación sale en inglés, debería estar en español para mejor experiencia con el usuario .

Ruta: [https://delystaging.herokuapp.com/business\\_types](https://delystaging.herokuapp.com/business_types)



Error: la opción de ver ( el perfil de cliente) en la lista de clientes no está habilitada. Ruta: [https://delystaging.herokuapp.com/admin/lista\\_de\\_clientes](https://delystaging.herokuapp.com/admin/lista_de_clientes)

la x +

S

**Pedido a restaurant la place** ✕

28 de Marzo de 2020 a las 06:07 PM  
Entregar en Calle 19 # 50 - 112a, marbella, Casa

**Estado del Pedido: Activo**

- Aprobando 06:07 PM
- En Preparación 06:08 PM
- En Camino
- Entregado

**Costo de Pedido**

**\$94.800**

Productos: \$90.500 + Domicilio: \$4.300

Tarjeta

**Información de Transacción**

Estado: payment\_not\_charged  
Respuesta de Pago Digital

**Información de Cliente**

miguel herrera  
3213546543  
cualpascual.07@gmail.com

**Información de Restaurante**

restaurant la place  
calle 1 N° 62-8A  
359456221 | 3123331212  
Edward

**Información de Domiciliario**

Recoger domicilio a las: 06:28 pm

Two red arrows point to the title 'Pedido a restaurant la place' and the delivery time 'Recoger domicilio a las: 06:28 pm'.

Lista de Clientes

Buscar  Todos los Valores

Nombre	Correo	Teléfono	Ver Perfil
juan manuel roa cala	juanmanuel125@gmail.com	300 6756569	Ver
miguel herrera	cualpascual.07@gmail.com	3213546543	Ver
Ivan Quintero	navi834@hotmail.com	3112445622	Ver
Julian Chavez	poisonfenix@hotmail.com		Ver
Juan David Gutierrez A	jeljuandavid@hotmail.com		Ver
Miguel Herrera	miguelhmonroy@hotmail.com		Ver
EDWARD HERNANDEZ	domiciliosdely1@gmail.com	3208168897	Ver
sergio chavez	hola123@fgdfd.com	32215	Ver
ivan torres	pihpapa2@yopmail.com	sdddeds	Ver
Pruebas dely	pruebas@gmail.com	3132694664	Ver
julipo	poi@hotmial.com	32255	Ver
Edward Hernandez	edwardsmaniaco@hotmail.com	3132694664	Ver
miguel h monroy	prueba@prueba.com	dssdr36565	Ver
dely pruebas	delypruebas@dely.com.co	313232323	Ver
Sebastian	sebaspachon94@gmail.com	3231231231	Ver

Error: la plataforma coloca 20 minutos de tiempo de recogida del domicilio sin importar el tiempo que el restaurante le ponga tiempos diferentes Ruta:

<https://delystaging.herokuapp.com/admin/panelpedidos>

## Apéndice D. informes generados por jira

Presentación de las diferentes actividades involucradas en el proyecto, a través de la herramienta jira. Donde se visualiza el informe generado por esta plataforma.

### Informe sprint 1

Proyectos / FLYKE / Informes

**Gráfica de trabajo pendiente en el sprint** [> Cómo leer este informe](#)

**Informe: Tablero Sprint 1** \* Incidencia añadida tras el inicio del sprint

**Registro de cambios del alcance** [Ver en el navegador de incidencias](#)

Fecha*	Clave :	Resumen :	Tipo de incidencia :	Epic :	Detalles del cambio de alcance :	Cambio en la estimación
2019-12-04	FLYKE-37*	Crear interfaz de manejo de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLL...	Incidencia añadida al sprint	-

**Incidencias incompletas** [Ver en el navegador de incidencias](#)

Clave :	Resumen :	Tipo de incidencia :	Epic :	Estado :	Persona asignada :	Puntos de historia
FLYKE-34	Crear la interfaz de inicio de sesión	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	-
FLYKE-35	Crear la interfaz de selección de categoría	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	-
FLYKE-36	Crear la interfaz de selección de producto	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	-
FLYKE-37	Crear interfaz de manejo de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	FINALIZADA	JC	-

## Informe sprint 2

Proyectos / FLYKE / Informes [Cómo leer este informe](#)

### Gráfica de trabajo pendiente en el sprint

**Informe: Tablero Sprint 2** \* Incidencia añadida tras el inicio del sprint

**Registro de cambios del alcance** [Ver en el navegador de incidencias](#)

Fecha *	Clave :	Resumen :	Tipo de incidencia :	Epic :	Detalles del cambio de alcance :	Cambio en la estimación
2020-01-08	FLYKE-37	Crear interfaz de manejo de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	La estimación ha cambiado de 4 a 2	4 → 2

Tu compromiso con el sprint ha disminuido en 2 puntos de historia  
Debido a cambios en el alcance: te quedan 49 puntos de historia para completar este sprint

**Incidencias incompletas** [Ver en el navegador de incidencias](#)

Clave :	Resumen :	Tipo de incidencia :	Epic :	Estado :	Persona asignada :	Puntos de historia
FLYKE-43	Implementar APIs en la vista de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	FINALIZADA	MH	8
FLYKE-42	Implementar Adiciones	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	12
FLYKE-39	Crear pantalla de inicio de sesión por PIN por parte de emp...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	9

Proyectos / FLYKE / Informes [Cómo leer este informe](#)

### Gráfica de trabajo pendiente en el sprint

FLYKE-43	Implementar APIs en la vista de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	FINALIZADA	MH	8
FLYKE-42	Implementar Adiciones	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	12
FLYKE-39	Crear pantalla de inicio de sesión por PIN por parte de emp...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	9

**Incidencias completadas** [Ver en el navegador de incidencias](#)

Clave :	Resumen :	Tipo de incidencia :	Epic :	Estado :	Persona asignada :	Puntos de historia
FLYKE-37	Crear interfaz de manejo de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	FINALIZADA	JC	2
FLYKE-34	Crear la interfaz de inicio de sesión	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	4
FLYKE-35	Crear la interfaz de selección de categoría	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	2
FLYKE-36	Crear la interfaz de selección de producto	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	1
FLYKE-40	Validar ingredientes antes de agregarlo a un pedido	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	5
FLYKE-41	Agregar un producto a un pedido directamente si no tiene ...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	6



## Informe Sprint 3

Proyectos / FLYKE / Informes

### Gráfica de trabajo pendiente en el sprint

[Cómo leer este informe](#)

**Informe: Tablero Sprint 3** \* Incidencia añadida tras el inicio del sprint

**Registro de cambios del alcance** [Ver en el navegador de incidencias](#)

Fecha	Clave	Resumen	Tipo de incidencia	Epic	Detalles del cambio de alcance	Cambio en la estimación
2020-01-29	FLYKE-53*	se debe mover de posición el botón de agregar e...	Error	MODULO DE FA...	Incidencia añadida al sprint	-
2020-01-31	FLYKE-54*	al asignar un pin ya creado para otro usuario, no h...	Error	MODULO DE CLI...	Incidencia añadida al sprint	-

**Incidencias incompletas** [Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-44	Implementar vista de pago de pedido	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	9
FLYKE-50	Implementar vista de mesas	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	4
FLYKE-52	Recargar el pedido con la información anterior si un pedido...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	6

**Incidencias completadas** [Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-39	Crear pantalla de inicio de sesión por PIN por parte de emp...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	1
FLYKE-42	Implementar Adiciones	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	2
FLYKE-43	Implementar APIs en la vista de empleados	<input checked="" type="checkbox"/> Tarea	MODULO DE CLIE...	FINALIZADA	MH	4
FLYKE-45	Corregir en el listado de pedido total de producto	<input type="checkbox"/> Error	MODULO DE FACT...	FINALIZADA	JC	2
FLYKE-46	Modificar tabla invoice	<input type="checkbox"/> Error	MODULO DE FACT...	FINALIZADA	SC	1
FLYKE-47	Eliminar los datos guardados en BD cuando cierra sesión	<input type="checkbox"/> Error	MODULO DE FACT...	FINALIZADA	JC	2
FLYKE-48	Guardar los ingredientes y extras en el invoice cuando se g...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	5
FLYKE-49	Agregar funcionalidad de buscador en ordenes	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	5
FLYKE-51	Implementar vista de selección de venues	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	5
FLYKE-53	se debe mover de posición el botón de agregar empleado	<input type="checkbox"/> Error	MODULO DE FACT...	FINALIZADA	MH	-
FLYKE-54	al asignar un pin ya creado para otro usuario, no hay mens...	<input type="checkbox"/> Error	MODULO DE CLIE...	FINALIZADA	MH	-

## Informe Sprint 4

Proyectos / FLYKE / Informes

## Gráfica de trabajo pendiente en el sprint

[> Cómo leer este informe](#)

## Registro de cambios del alcance

[Ver en el navegador de incidencias](#)

Fecha	Clave	Resumen	Tipo de incidencia	Epic	Detalles del cambio de alcance	Cambio en la estimación
2020-02-18	FLYKE-65*	Implementar actualización forzada en la app	<input checked="" type="checkbox"/> Tarea		Incidencia añadida al sprint	-
2020-02-20	FLYKE-65	Implementar actualización forzada en la app	<input checked="" type="checkbox"/> Tarea		Se ha añadido la estimación de 2	- → 2

Tu compromiso con el sprint ha aumentado en 2 puntos de historia  
Debido a cambios en el alcance: te quedan 48 puntos de historia para completar este sprint

## Incidencias incompletas

[Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-58	Crear vista de categorías para administrador	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	7
FLYKE-59	Crear vista de reportes	<input checked="" type="checkbox"/> Tarea		TAREAS POR HACER		3
FLYKE-60	Crear vista de cierre de caja	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	6

Proyectos / FLYKE / Informes

## Gráfica de trabajo pendiente en el sprint

[> Cómo leer este informe](#)

## Incidencias completadas

[Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-44	Implementar vista de pago de pedido	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	2
FLYKE-52	Recargar el pedido con la información anterior si un pedido...	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	MH	6
FLYKE-50	Implementar vista de mesas	<input checked="" type="checkbox"/> Tarea	MODULO DE FACT...	FINALIZADA	JC	1
FLYKE-55	Implementar SDK de firebase	<input checked="" type="checkbox"/> Tarea		FINALIZADA	SC	2
FLYKE-56	Ajustar vista de módulos con la llamada del API	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	3
FLYKE-57	Realizar correcciones a la vista de categorías	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	8
FLYKE-61	Realizar correccion a la vista de inicio de sesión	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	2
FLYKE-62	Realizar correccion a la vista de PIN	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	2
FLYKE-63	Realizar correccion a la vista de módulos	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	2
FLYKE-64	Realizar corrección de vista de empleados	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	2
FLYKE-65	Implementar actualización forzada en la app	<input checked="" type="checkbox"/> Tarea		FINALIZADA	IT	2

## Informe Sprint 5

Proyectos / FLYKE / Informes

### Gráfica de trabajo pendiente en el sprint

[Cómo leer este informe](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-58	Crear vista de categorías para administrador	✓ Tarea		FINALIZADA	JC	5
FLYKE-67	Realizar impresión de ordenes	✓ Tarea		FINALIZADA	JC	8
FLYKE-68	Crear módulo de gastos/ingresos	✓ Tarea		TAREAS POR HACER		5
FLYKE-70	Crear módulo de devoluciones	✓ Tarea		TAREAS POR HACER		3
FLYKE-71	Ajustar toma pedidos a los diferentes tipos de negocio	✓ Tarea		EN CURSO	SC	6
FLYKE-72	Ajustar actividad vista de pagos	✓ Tarea		FINALIZADA	JC	10
FLYKE-73	Crear vista de créditos dely	✗ Error		EN CURSO	JC	6

**Incidencias completadas** [Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-60	Crear vista de cierre de caja	✓ Tarea		FINALIZADA	MH	3
FLYKE-66	Ajustar vista de empleados para perfiles bloqueados	✓ Tarea		FINALIZADA	JC	3
FLYKE-69	Corregir cerrar sesión en la vista de módulos	✓ Tarea		FINALIZADA	MH	2

## Informe Sprint 6

Proyectos / FLYKE / Informes

### Gráfica de trabajo pendiente en el sprint

[Cómo leer este informe](#)

**Informe: Tablero Sprint 6** \* Incidencia añadida tras el inicio del sprint

**Registro de cambios del alcance** [Ver en el navegador de incidencias](#)

Fecha	Clave	Resumen	Tipo de incidencia	Epic	Detalles del cambio de alcance	Cambio en la estimación
2020-03-27	FLYKE-77*	Regresión Lanzamiento Dely Miguel	✓ Tarea		Incidencia añadida al sprint	-
2020-03-27	FLYKE-78*	Regresión Lanzamiento Dely Roa	✓ Tarea		Incidencia añadida al sprint	-

**Incidencias incompletas** [Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-73	Crear vista de créditos dely	✗ Error		EN CURSO	JC	6
FLYKE-70	Crear módulo de devoluciones	✓ Tarea		TAREAS POR HACER		3
FLYKE-71	Ajustar toma pedidos a los diferentes tipos de negocio	✓ Tarea		EN CURSO	SC	6
FLYKE-72	Ajustar actividad vista de pagos	✓ Tarea		FINALIZADA	JC	10
FLYKE-68	Crear módulo de gastos/ingresos	✓ Tarea		TAREAS POR HACER		5
FLYKE-76	Implementar funcionalidad de impresión con respuestas del API	✓ Tarea		FINALIZADA	MH	-

**Incidencias completadas** [Ver en el navegador de incidencias](#)

Clave :	Resumen :	Tipo de incidencia :	Epic :	Estado :	Persona asignada :	Puntos de historia
FLYKE-67	Realizar impresión de ordenes	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	8
FLYKE-58	Crear vista de categorías para administrador	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	5
FLYKE-77	Regresión Lanzamiento Dely Miguel	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	-
FLYKE-78	Regresion Lanzamiento Dely Roa	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	-

## Sprint 7

Proyectos / FLYKE / Informes [> Cómo leer este informe](#)

**Gráfica de trabajo pendiente en el sprint**

**Informe: Tablero Sprint 7** \* Incidencia añadida tras el inicio del sprint

**Registro de cambios del alcance** [Ver en el navegador de incidencias](#)

Fecha :	Clave :	Resumen :	Tipo de incidencia :	Epic :	Detalles del cambio de alcance :	Cambio en la estimación
2020-04-24	FLYKE-81*	Crear fragmento de tabs de pago	<input checked="" type="checkbox"/> Tarea		Incidencia añadida al sprint	-
2020-05-04	FLYKE-82*	Rompe filas y foodtruck	<input checked="" type="checkbox"/> Error		Incidencia añadida al sprint	-

**Incidencias incompletas** [Ver en el navegador de incidencias](#)

Clave :	Resumen :	Tipo de incidencia :	Epic :	Estado :	Persona asignada :	Puntos de historia
FLYKE-73	Crear vista de créditos dely	<input checked="" type="checkbox"/> Error		EN CURSO	JC	6
FLYKE-71	Ajustar toma pedidos a los diferentes tipos de negocio	<input checked="" type="checkbox"/> Tarea		EN CURSO	SC	2
FLYKE-79	Crear vista de pago único	<input checked="" type="checkbox"/> Tarea		EN CURSO	SC	-
FLYKE-80	Crear vista de pago en partes	<input checked="" type="checkbox"/> Tarea		EN CURSO	JG	-
FLYKE-82	Rompe filas y foodtruck	<input checked="" type="checkbox"/> Error		EN CURSO	SC	-

Proyectos / FLYKE / Informes

### Gráfica de trabajo pendiente en el sprint

[Cómo leer este informe](#)

Incidencias completadas [Ver en el navegador de incidencias](#)

Clave	Resumen	Tipo de incidencia	Epic	Estado	Persona asignada	Puntos de historia
FLYKE-76	Implementar funcionalidad de impresión con respuestas del API	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	3
FLYKE-72	Ajustar actividad vista de pagos	<input checked="" type="checkbox"/> Tarea		FINALIZADA	JC	10
FLYKE-81	Crear fragmento de tabs de pago	<input checked="" type="checkbox"/> Tarea		FINALIZADA	MH	-

Incidencias completadas fuera del sprint

## Apéndice E. vistas del proyecto FLYKE

### Vistas del proyecto.

A continuación se muestran algunas interfaces del proyecto FLYKE que fue el pilar principal en el desarrollo de la pasantía

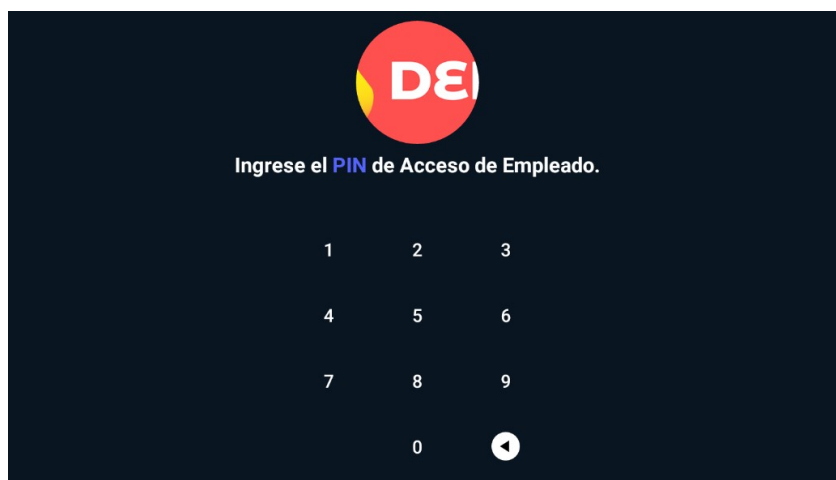


Imagen 8. Interfaz de inicio con PIN

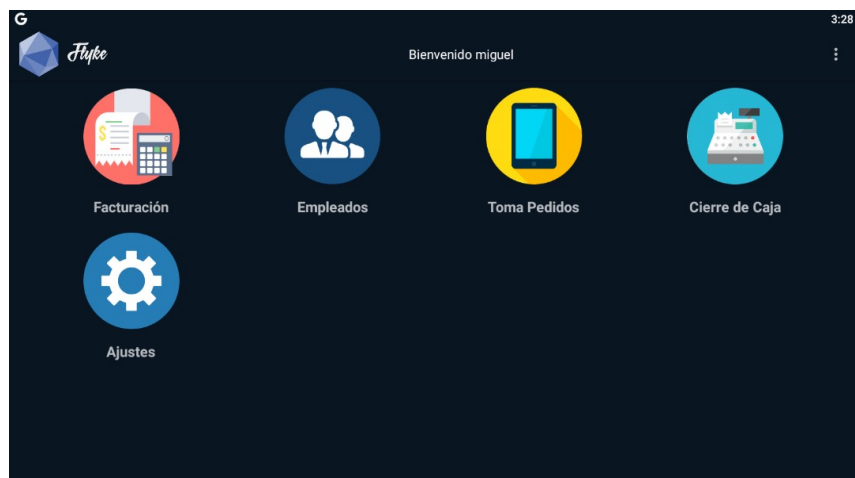


Imagen 9. Interfaz de selección de módulo

The screenshot shows the Facturación (Billing) module interface. It features a table with columns for Mesa (Table), Hora (Time), Estado (Status), and Total. Each row represents an order with associated icons for cart, print, and receipt.

Mesa	Hora	Estado	Total			
Altillo 1 Orden: #156	08:00 am	No Disponible	111.400 COP			
Altillo 2 Orden: #159	08:00 am	No Disponible	66.400 COP			
Mesa 1 Orden: #158	08:00 am	No Disponible	42.000 COP			
Mesa 2 Orden: #157	08:00 am	No Disponible	43.400 COP			
Mesa 3 Orden: #160	08:00 am	No Disponible	39.400 COP			
Mesa 4 Orden: #162	08:00 am	No Disponible	57.000 COP			
Mesa 5						

Imagen 10. Interfaz de módulo de facturación

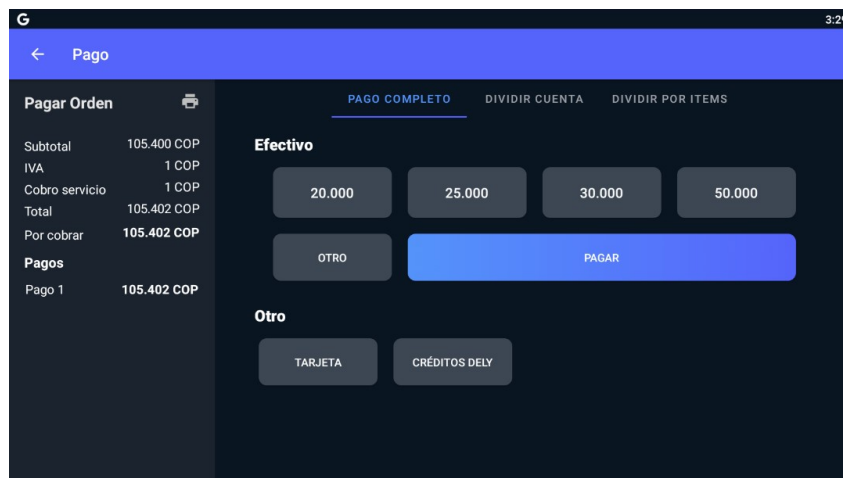


Imagen 11. Interfaz de pago de factura

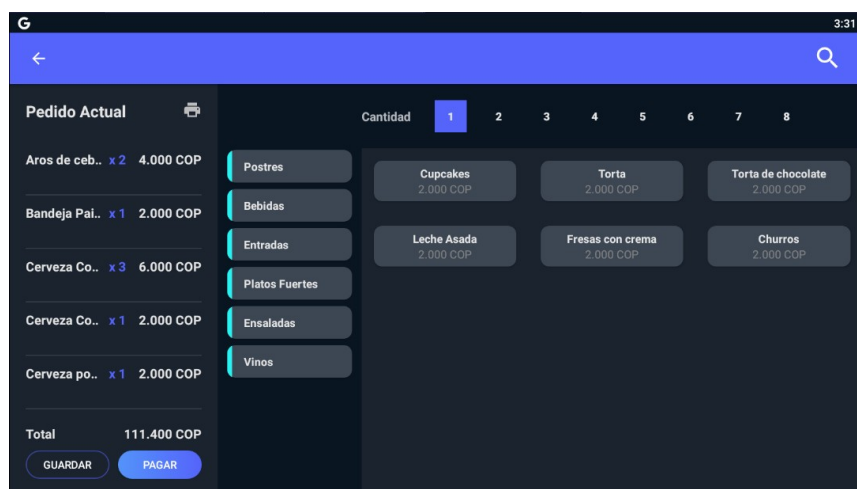
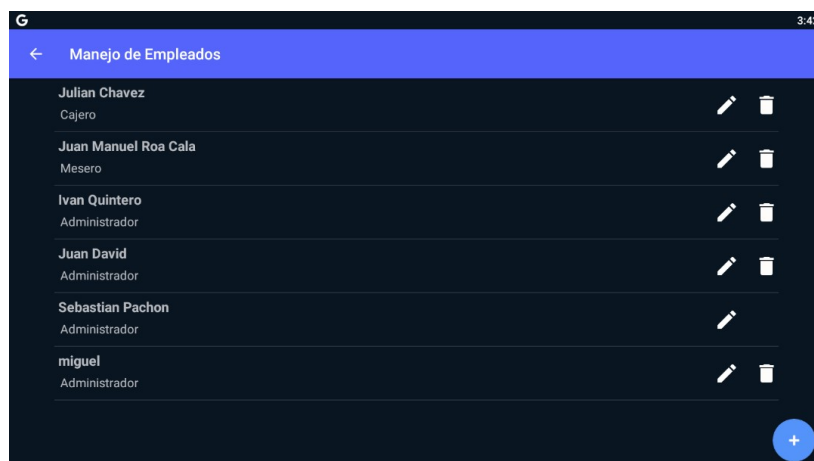


Imagen 12. Interfaz de facturación de pedidos





*Imagen 13. Interfaz de manejo de empleados*