

**Version 1.0**

DESCARGARPORTADAS.COM



**Plan de Pruebas**

2020

**Aplicación Móvil Tacna Food and Drinks**

Revisión Histórica

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 28/11/2020 | 1.0 | Primera realización del Plan de Pruebas | Marko Rivas Rios  Ronald Ordoñez Quilli |

1. Introducción

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 requisitos que deben se 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 “Tacna Food and Drinks”.

Este documento, específicamente:

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

1. **Alcance**

Este documento de Plan de Pruebas se convierte en una guia para desarrollar de forma organizada las diferentes actividades que se realizarán en el proceso de las pruebas durante del desarrollo del proyecto “Tacna Food and Drinks”.

1. **Definiciones, acrónimos y abreviaturas**

* Plan de pruebas: Describe todos los métodos que se utilizaran para verificar que el aplicativo satisface la especificación del producto y las necesidades del cliente. Incluye los objetivos de calidad, necesidades de recursos, cronograma, métodos, etc.
* Casos de prueba: Lista los ítems específicos que serán probados y describe los pasos detallados que serán seguidos para verificar el aplicativo.
* Reporte de pruebas: Describen los problemas o errores encontrados al ejecutar los casos de prueba.

1. **Referencias**

* Alcance del proyecto
* Documento de especificación de requisitos
* Casos de uso
* Descripción de casos de uso
* Diagrama de clases

1. **Herramientas para las pruebas**

Como herramienta principal se utilizará jUnit, el cual es un marco simple para escribir pruebas unitarias.