 DOCUMENTACIÓN DE ENTREGA

SERVICIO DE EMPLEO MOVIL

IMPLEMENTACIÓN DE APLICACIONES WEB Y MÓVILES BAJO UN MODELO DE DESARROLLO DE APLICACIONES ÁGIL Y PARTICIPATIVO PARA EL PROGRAMA AGENDA DE CONECTIVIDAD – ESTRATEGIA GOBIERNO EN LÍNEA, EN EL MARCO DEL PLAN VIVE DIGITAL

Dirección de Gobierno en línea

@República de Colombia – Derechos Reservados

Bogotá D.C, Noviembre del 2014

FORMATO PRELIMINAR AL DOCUMENTO

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Título: | SERVICIO DE EMPLEO MOVIL | | | | |
| Fecha elaboración:  aaaa-mm-dd | 2014-11-23 | | | | |
| Sumario: | El presente documento tiene como objetivo documentar el planteamiento y recursos para desarrollar el reto Servicio de empleo movil, con el fin de definir claramente la problemática a solicitar con la aplicación. | | | | |
| Palabras Claves: | Empleo, Filtros, Webservice | | | | |
| Formato: | DOC | | Lenguaje: | | Español |
| Dependencia: | Ministerio de Tecnologías de la Información y las Comunicaciones: Fondo de Tecnologías de la Información y las Comunicaciones | | | | |
| Código: | MOV-447-GELMOV-IT4-DO-Documentación\_ | Versión: | 1.0 | Estado: | Generado |
| Categoría: | NA | | | | |
| Autor (es): | Equipo  Consorcio Software - 2012 | | Firmas |  | |
| Revisó: | **Diego Fernando Rocha Arango**  Consultor Diseño e Innovación / Dirección de Gobierno en Línea  **Héctor Oswaldo Bonilla**  Consultor Diseño e Innovación / Dirección de Gobierno en Línea  **Santiago García Davis**  Coordinador de Diseño e Innovación  Dirección de Gobierno en Línea | |  |  | |
| Aprobó: | **Johanna Pimiento**  Directora de Gobierno en Línea | |  |  | |
| Información Adicional: | NA | | | | |
| Ubicación: | El archivo magnético asociado al documento está localizado en el repositorio del proyecto en la siguiente ruta: <http://bog.ospinternational.com:8889/browse/MOV-408> | | | | |

CONTROL DE CAMBIOS

| **VERSIÓN** | **FECHA** | **No. SOLICITUD** | **RESPONSABLE** | **DESCRIPCIÓN** |
| --- | --- | --- | --- | --- |
| 1.0 | 23/11/2014 | No aplica | Consorcio Software 2012 | Creación del documento |

TABLA DE CONTENIDO

[DERECHOS DE AUTOR 9](#_Toc404244636)

[CRÉDITOS 11](#_Toc404244637)

[1. AUDIENCIA 12](#_Toc404244638)

[2. INTRODUCCIÓN 13](#_Toc404244639)

[3. RE - PLANTEAMIENTO DE PROBLEMA 15](#_Toc404244640)

[4. RE - DESCRIPCIÓN DEL RETO 16](#_Toc404244641)

[5. RE - REQUERIMIENTOS FUNCIONALES 17](#_Toc404244642)

[5.1 LISTADO DE INSUMOS 17](#_Toc404244643)

[5.3 PUNTOS DE VENTA 18](#_Toc404244644)

[5.6 SOY VENDEDOR 18](#_Toc404244645)

[6. RE - REQUERIMIENTOS NO FUNCIONALES 20](#_Toc404244646)

[7. RE - REQUERIMIENTOS DE SEGURIDAD 23](#_Toc404244647)

[8. RE - FUENTES DE DATOS 25](#_Toc404244648)

[8.1 WEB SERVICES APLICACIÓN: 25](#_Toc404244649)

[9. - DESCRIPCIÓN DE LA APLICACIÓN DI 27](#_Toc404244650)

[10. DI - ARQUITECTURA GENERAL 28](#_Toc404244651)

[11. DI - MODELO CONCEPTUAL 29](#_Toc404244652)

[12. DI - VISTA LÓGICA (DIAGRAMA DE COMPONENTES) 31](#_Toc404244653)

[12.1 VISTA LÓGICA GENERAL 31](#_Toc404244654)

[12.2 COMPONENTE: APLICACIÓN MÓVIL 32](#_Toc404244655)

[12.2.1PROPÓSITO 32](#_Toc404244656)

[13. DI - MODELO DE DATOS 34](#_Toc404244657)

[14. DI - VISTA DE IMPLEMENTACIÓN DE INTERFACES ENTRE COMPONENTES 35](#_Toc404244658)

[14.1 VISTA DE DESPLIEGUE 35](#_Toc404244659)

[15. DI - VISTA DE INTEGRACIÓN CON SISTEMAS EXTERNOS 38](#_Toc404244660)

[16. DI - VISTA DE PARAMETRIZACIÓN DEL SISTEMA 39](#_Toc404244661)

[17. DI - HISTORIAS DE USUARIO 57](#_Toc404244662)

[18. EE - ESTIMACIÓN DE ESFUERZO 64](#_Toc404244663)

[19. PPA - PLAN DE PRUEBAS DETALLADO 68](#_Toc404244664)

[20. PPA - TIPOS DE PRUEBA 72](#_Toc404244665)

[21. PPA - AMBIENTE DE PRUEBAS 75](#_Toc404244666)

[22. CA - CRITERIOS DE ACEPTACIÓN 80](#_Toc404244667)

[23. CA - CASOS DE PRUEBA DE LA APLICACIÓN 82](#_Toc404244668)

[24. IP - HISTORIAS DE USUARIO PROBADAS 95](#_Toc404244669)

[25. IP - REGISTRO Y CLASIFICACIÓN DE INCIDENCIAS 98](#_Toc404244670)

[26. IP - GESTIÓN DE INCIDENCIAS 100](#_Toc404244671)

[27. MI - PRE-REQUISITOS 111](#_Toc404244672)

[28. MI - PREPARACIÓN DE ADMINISTRADORES DE LA APLICACIÓN Y ARCHIVOS DE INSTALACIÓN 114](#_Toc404244673)

[29. MI - PROCESO DE CREACIÓN Y CONFIGURACIÓN DE BASE DE DATOS 126](#_Toc404244674)

[30. MI - INSTALACIÓN DE LA APLICACIÓN 130](#_Toc404244675)

[31. MI - VERIFICAR INSTALACIÓN 136](#_Toc404244676)

[32. MI - PROBLEMAS FRECUENTES 137](#_Toc404244677)

[33. MI - INSTALADORES 138](#_Toc404244678)

[34. MU - OBJETIVOS DEL SISTEMA 140](#_Toc404244679)

[35. MU - REQUISITOS DEL SISTEMA 141](#_Toc404244680)

[36. MU - DESCRIPCIÓN DEL SISTEMA 142](#_Toc404244681)

[36.1 ADMINISTRADOR 142](#_Toc404244682)

[36.2 USUARIO FINAL 142](#_Toc404244683)

[37. MU - MANUAL DE USO 143](#_Toc404244684)

[37.1 ABRIR LA APLICACIÓN 143](#_Toc404244685)

[37.2 MENÚ PRINCIPAL 144](#_Toc404244686)

[37.3 LISTADO DE INSUMOS 145](#_Toc404244687)

[37.4 COMPARATIVO DE PRECIOS 147](#_Toc404244688)

[37.5 REPORTE DE PRECIOS 148](#_Toc404244689)

[37.6 PUNTOS DE VENTA 149](#_Toc404244690)

[37.7 CONSEJOS PRÁCTICOS 150](#_Toc404244691)

[37.8 BOLETINES INFORMATIVOS 151](#_Toc404244692)

[37.9 REGISTRAR PUNTO DE VENTA 151](#_Toc404244693)

[38. MU - MANUAL DE OPERACIÓN 153](#_Toc404244694)

[39. TERMINOLOGÍA 157](#_Toc404244695)

[40. ANEXOS 160](#_Toc404244696)

TABLA DE ILUSTRACIONES

[Ilustración 1. Arquitectura General 30](file:///C:\Users\Mariluz\Desktop\MOV-447-GELMOV-IT3-DO-Documentacion-Agro%20Insumos%20Colombia.docx#_Toc397959767)

[Ilustración 2. Diagrama de Modelo Conceptual 32](#_Toc397959768)

[Ilustración 3. Diagrama de Vista Lógica 34](#_Toc397959769)

[Ilustración 5. Modelo de datos 36](#_Toc397959770)

[Ilustración 4 - Vista de despliegue 39](#_Toc397959771)

[Ilustración 7. Vista de Integración con sistemas externos. 41](#_Toc397959772)

[Ilustración 8. Tiempo de Resolución. 114](#_Toc397959773)

[Ilustración 9.Incidencias creadas vs. Resueltas 114](#_Toc397959774)

[Ilustración 10. Descargar Github for Windows 121](#_Toc397959775)

[Ilustración 11.Repositorio local GitHub 122](#_Toc397959776)

[Ilustración 12. Formulario de creación de repositorio. 123](#_Toc397959777)

[Ilustración 13. Nuevo repositorio creado 124](#_Toc397959778)

[Ilustración 14. Lista de archivos para cargar en el repositorio. 125](#_Toc397959779)

[Ilustración 15. Certificados identificadores y perfiles. 126](#_Toc397959780)

[Ilustración 16. IOS Apps 127](#_Toc397959781)

[Ilustración 17. Registrar UDID de dispositivos IOS 127](#_Toc397959782)

[Ilustración 18. Formulario de registro -Nuevo UDID 128](#_Toc397959783)

[Ilustración 19.Información del dispositivo 129](#_Toc397959784)

[Ilustración 20. Registro completo 129](#_Toc397959785)

[Ilustración 21.Descarga del aplicativo 136](#_Toc397959786)

[Ilustración 22. Pantalla de instalación del aplicativo. 137](#_Toc397959787)

[Ilustración 23. Mensaje de aplicación instalada 138](#_Toc397959788)

[Ilustración 24. Arrastrar archivo .ipa a iTunes 139](#_Toc397959789)

[Ilustración 25. Ver información del dispositivo 139](#_Toc397959790)

[Ilustración 26.. Instalar aplicación desde iTunes 140](#_Toc397959791)

[Ilustración 27.Logo del Aplicativo 149](#_Toc397959792)

[Ilustración 28.Inicio de la aplicación con el logo 150](#_Toc397959793)

DERECHOS DE AUTOR

A

menos que se indique de forma contraria, el derecho de copia del texto incluido en este documento es del Gobierno de la República de Colombia. Se puede reproducir gratuitamente en cualquier formato o medio sin requerir un permiso expreso para ello, bajo las siguientes condiciones:

1. El texto particular no se ha indicado como excluido y por lo tanto no puede ser copiado o distribuido.
2. La copia no se hace con el fin de distribuirla comercialmente.
3. Los materiales se deben reproducir exactamente y no se deben utilizar en un contexto engañoso.
4. Las copias serán acompañadas por las palabras "copiado/distribuido con permiso de la República de Colombia. Todos los derechos reservados."
5. El título del documento debe ser incluido al ser reproducido como parte de otra publicación o servicio.

Si se desea copiar o distribuir el documento con otros propósitos, se debe solicitar el permiso entrando en contacto con La Dirección de Gobierno en Línea del Ministerio de Tecnologías de la Información y las Comunicaciones de la República de Colombia.

CRÉDITOS

Dentro de la ejecución del contrato celebrado entre el Fondo de Tecnologías de la Información y las Comunicaciones y el Consorcio Software - 2012 para

cumplir con el alcance previsto en el proyecto: **IMPLEMENTACIÓN DE APLICACIONES WEB Y MÓVILES BAJO UN MODELO DE DESARROLLO DE APLICACIONES ÁGIL Y PARTICIPATIVO PARA EL PROGRAMA AGENDA DE CONECTIVIDAD – ESTRATEGIA GOBIERNO EN LÍNEA, EN EL MARCO DEL PLAN VIVE DIGITAL**, se ha generado el presente documento, el cual ha sido elaborado por profesionales del Consorcio Software – 2012, siguiendo los estándares establecidos por la Dirección de Gobierno en Línea.

Todos los documentos son revisados y aprobados por los consultores y profesionales de la Dirección de Gobierno en Línea.

1. AUDIENCIA

E

ste documento está dirigido a los miembros del equipo de la Dirección de Gobierno en Línea, Servicio de empleo y el Consorcio Software – 2012, que participan en la definición del reto Servicio de empleo movil, el cual se incluye en el proyecto **IMPLEMENTACIÓN DE APLICACIONES WEB Y MÓVILES BAJO UN MODELO DE DESARROLLO DE APLICACIONES ÁGIL Y PARTICIPATIVO PARA EL PROGRAMA AGENDA DE CONECTIVIDAD – ESTRATEGIA GOBIERNO EN LÍNEA, EN EL MARCO DEL PLAN VIVE DIGITAL.**

1. INTRODUCCIÓN

L

a aplicación de entrega para el reto Servicio de empleo móvil busca generar un canal de comunicación rápido y eficiente entre la ciudadanía y el servicio de empleo.

ANÁLISIS DE REQUERIMIENTOS

1. RE - PLANTEAMIENTO DE PROBLEMA

El servicio de empleo en el pais, dispone de información sobre empleos disponibles en el territorio nacional para diferentes perfiles y caracteristicas. Se requiere facilitar el acceso a dicha información por parte de los ciudadanos y los empledores de una manera dinamica y rapida, aprovechando las tecnologias de la información que se disponen en la actualidad.

1. RE - DESCRIPCIÓN DEL RETO

Cómo permitir a las personas y a los empleadores comunicarse entre si, facilitar la publicación y consulta de la información del empleo en el país.

1. RE - REQUERIMIENTOS FUNCIONALES

El sistema debe contar con las siguientes opciones:

5.1 Oportunidades de empleo (Móvil y Web)

Este módulo permite la búsqueda y consulta de oportunidades de empleo, existe tanto en la aplicación móvil como en el sitio web, la búsqueda de oportunidades se realiza según filtros como: Tipo de oportunidad, Departamento, Municipio, Salario, Experiencia requerida, Nivel educativo; además e una o varias palabras claves.

Desde este módulo se puede acceder además a visualizar las ofertas de empleo buscadas en un mapa del territorio nacional.

**5.2 Mis vacantes favoritas**

Esta funcionalidad estará disponible en la aplicación móvil y permite al usuario seleccionar diferentes ofertas de empleo como favoritas y tener información de estas a primera mano sin necesidad de realizar búsquedas.

5.3 Sección de empleador

1. Registrar oferta (Móvil y Web)

El empleador tiene la posibilidad de registrar nuevas ofertas de empleo despues de iniciar sesión, tanto en la aplicación móvil como en el sitio web.

1. **Editar oferta (Móvil y Web)**

Este módulo permite editar la información de las oportunidades de empleo registradas en el sistema.

**5.4 Preguntas frecuentes (Móvil y Web)**

Esta funcionalidad es una ayuda de primera mano para los usuarios, contiene preguntas que son de interés general y brindan una orientación sobre el uso del sistema y el empleo en general.

**5.5 Acerca de**

La aplicación permitirá visualizar información de los desarrolladores del producto, la entidad y mentores acompañantes del proceso.

1. RE - REQUERIMIENTOS NO FUNCIONALES

La aplicación debe cumplir con los siguientes requerimientos no funcionales:

* **Estándares de desarrollo**

Las aplicaciones deben ser desarrolladas para funcionar correctamente en los sistemas operativos Android en sus versiones 2.3.3 a 2.3.7 – Gingerbread, 4.0 Ice Cream Sandwich y 4.1 a 4.2 Jelly Bean; e IOS en sus versiones 5.1, 6.0 y 6.1.

Se debe cumplir con unos estándares mínimos de calidad en cuanto desarrollo y codificación. Estos estándares se encuentran especificados en el Anexo 1 de este documento.

* **Facilidad de instalación**

Las aplicaciones móviles deben tener facilidad de instalarse desde tiendas de aplicaciones según los sistemas operativos en los dispositivos.

* **Lenguajes de programación**

Las aplicaciones deben ser desarrolladas en los siguientes lenguajes de programación:

Java para aplicaciones Android

Objective-C o HTML5: para la plataforma iOS

* **Tiempos de respuesta**

Los tiempos de respuesta y funcionamiento de las aplicaciones que utilicen Servicios Web dependerán de la disponibilidad y desempeño tanto del esquema de conexión del móvil (3G, HSDPA, WiFi) como del servicio mismo ofrecido por la entidad respectiva. Las soluciones móviles deberán contemplar manejo de error cuando no exista disponibilidad o haya errores por timeout en los servicios de las entidades.

* **Tiempos de respuesta de arranque de las aplicaciones**

Es el tiempo transcurrido desde el momento en que se selecciona la aplicación en el dispositivo hasta que aparece la primera pantalla de la aplicación. Este tiempo está sujeto al uso de memoria del dispositivo que esté ejecutando la aplicación en un momento determinado, así como a la capacidad de procesamiento y memoria de dicho dispositivo.

* **Mantenibilidad**

Se debe permitir modificar los textos incluidos dentro de las aplicaciones de manera estática, así como las constantes relaciones a la conexión y autenticación de los Servicios Web. Para esto, el código fuente de las aplicaciones contará con archivos estrictamente dedicados a este fin, que pueden ser de los siguientes tipos:

XML para manejo de textos en ambiente nativo.

JSON para manejo de textos en ambiente híbrido.

Clases estáticas con parámetros constantes de conexión.

* **Manejo de resoluciones de pantalla**

La aplicación debe manejar las diferentes resoluciones de pantalla de manera automática. Es decir, si la aplicación se instala en una tableta o un celular, la pantalla debe organizarse automáticamente a las dimensiones del dispositivo.

* **Permisos de aplicación**

La aplicación, al momento de su instalación, debe solicitar únicamente aquellos permisos que sean indispensables para su funcionamiento.

1. RE - REQUERIMIENTOS DE SEGURIDAD

La seguridad en la aplicación estará fundamentada en el proyecto OWASP, con el fin de mitigar riesgos de seguridad concernientes a : Integralidad

Disponibilidad, Trazabilidad y Confidencialidad. La aplicación debe brindar

seguridad en cuanto a los siguientes riesgos:

**Integridad:**

* Transmitir información sensible sin encriptar.
* Descarga indebida de información.

**Disponibilidad:**

1. Almacenar Incontroladamente la Información.
2. Enviar mensajes SMS o Email incontroladamente.
3. Ejecución incontrolada de bucles.
4. Activación incontrolada de dispositivos Bluetooth, cámara, GPS, etc.

**Trazabilidad:**

* Ejecución Nativa de Funciones.
* Borrado de la Información

**Confidencialidad**

* Geolocalización del dispositivo.
* Utilización indebida de los contactos del dispositivo.

1. RE - FUENTES DE DATOS

****8.1 WEB SERVICES APLICACIÓN**:**

La aplicación cuenta con diversos web services, los cuales almacenan y retornan la información relacionada necesaria para trabajar con la aplicación.

Este web service tiene las siguientes características:

* Sistema Operativo, windows server 2008
* Servidor de Aplicaciones, internet information service
* Base de Datos, sql server
* Asp.net con servicios SOAP

DISEÑO

1. - DESCRIPCIÓN DE LA APLICACIÓN

La aplicación móvil para dar solución al reto Servicio de empleo movil, planteado por la misma entidad, durante la convocatoria móvil celebrada en la ciudad de Bogotá, permite consultar a todo tipo de usuarios las ofertas de empleo existentes a lo largo del territorio nacional categorizadas por diferentes filtros y permitiendo la busqueda especifica según las necesidades del usuario.

1. DI - ARQUITECTURA GENERAL

La arquitectura de la aplicación está diseñada bajo un modelo cliente-servidor básico, en el cual el dispositivo móvil actúa como cliente que recibe

información proporcionada por un servicio web. Para que la aplicación móvil, este

debe tener un sistema operativo Android 4.1 o superior, o sistema operativo iOS 5

o superior, el sitio web usa también los webservices construidos y requiere de un navegador web Chrome o Mozilla Firefox.

La aplicación está desarrollada en lenguaje nativo tanto IOS como ANDROID

Web Service Servicio de empleo



Android 2.3.2 o superior

iOS 5.X o superior

Ilustración . Arquitectura General

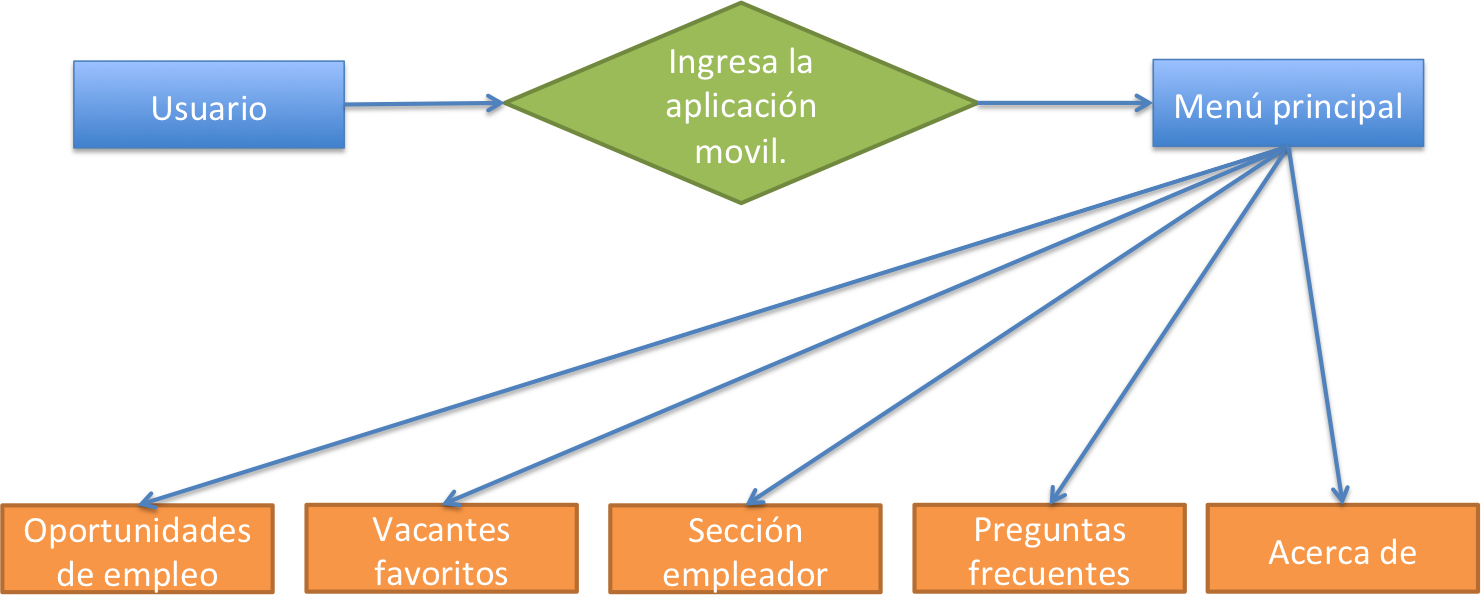
1. DI - MODELO CONCEPTUAL

D e conformidad con la política de funcionamiento del Estado con soporte en las TIC’s y desde la perspectiva de Gobierno Abierto, el Plan Vive Digital establece el desarrollo de aplicaciones que apoyen los procesos de las entidades públicas que le permitan a los ciudadanos obtener información útil, actualizada, clara y precisa.

De acuerdo con lo anterior se plantea el desarrollo de una aplicación móvil que permita la consulta de la oferta de empleo existente en el país, a las cuales todos los ciudadanos pueden aplicar.

El Modelo Conceptual de la aplicación SERVICIO DE EMPLEO MOVIL, tiene como fin reflejar la dimensión del trabajo a realizar desde una perspectiva de alto nivel, de tal manera que los interesados y los roles involucrados puedan tener un marco de referencia que les permita tener conocimiento sobre la estructura y funcionalidad de la aplicación.

La aplicación funciona como canal de comunicación donde los ciudadanos en todo el territorio nacional pueden realizar una consulta y busqueda de ofertas de empleo.



**ENTIDAD**

DPS

Ilustración 2. Diagrama de Modelo Conceptual

**ENTIDAD**

1. DI - VISTA LÓGICA (DIAGRAMA DE COMPONENTES)

La vista lógica general es una representación de la estructura de componentes, en donde se describen las piezas de software fundamentales para

la arquitectura destacando el papel que desempeñan dentro del diseño y la forma

que inter-operan con los otros elementos.

* 1. VISTA LÓGICA GENERAL

A continuación se describen los componentes de la vista lógica de la aplicación, una aplicación móvil para sistemas operativos Android y iOS y un sitio web que permite visualizar la oferta de empleo en Colombia.

Sitio web

Servicios web

Aplicación Móvil para iOS

Aplicación Móvil para Android

Fuente de Información

Cliente Móvil

Ilustración 3. Diagrama de Vista Lógica

12.2 COMPONENTE: APLICACIÓN MÓVIL

12.2.1PROPÓSITO

Este componente consulta los datos retornados por los servicios web de la comisión nacional del servicio civil a través de una interfaz gráfica que permite ordenar la información y presentarla a los usuarios finales de una manera amigable.

**12.2.2 COMPONENTE: SERVICIOS**

**12.2.3PROPÓSITO**

Este componente contiene el conjunto de servicios web que son consumidos por la aplicación móvil, de modo que la información acerca de la oferta de empleo, esté actualizada para su consulta y por lo tanto sea correcta la difusión hacia los usuarios finales.

**12.2.4 COMPONENTE: SITIO WEB**

**12.2.2.5 PROPÓSITO**

Este componente permite visualizar la información por medio de un navegador web desde cualquier lugar con acceso internet, consulta los datos retornados por los servicios web de la comisión nacional del servicio civil a través de una interfaz gráfica que permite ordenar la información y presentarla a los usuarios finales de una manera amigable.

1. DI - MODELO DE DATOS

Diagrama de Clases donde se muestra la estructura lógica de la aplicación, en la parte de almacenamiento de información.

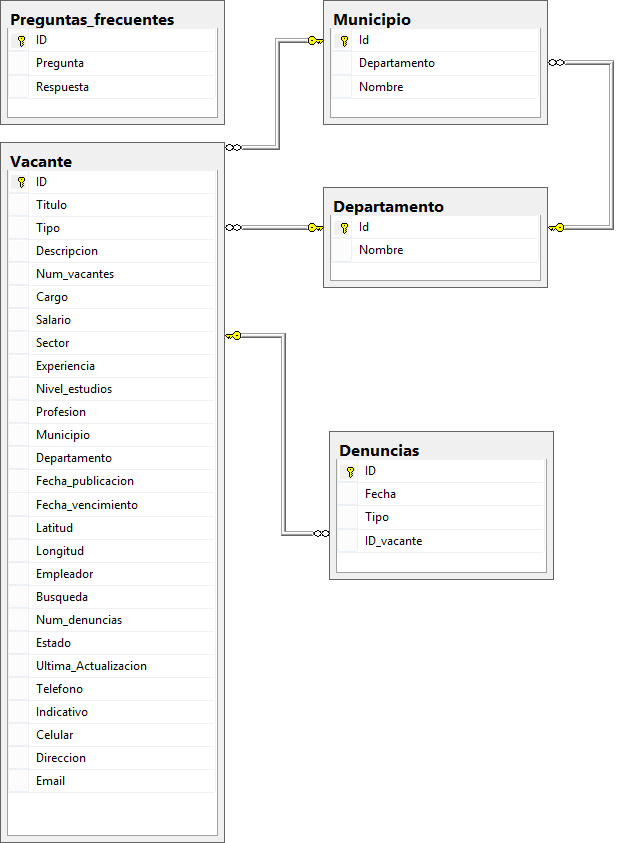


Ilustración 5. Modelo de datos

14. DI - VISTA DE IMPLEMENTACIÓN DE INTERFACES ENTRE COMPONENTES

La Vista de implementación comprende los artefactos o componentes necesarios para ensamblar la aplicación. Muestra la forma en que los

componentes descritos en la vista lógica se encuentran ubicados físicamente.

* 1. VISTA DE DESPLIEGUE

Los componentes del sistema están conformados por el componente móvil y el componente servidor que la Entidad pone a disposición. La aplicación interactúa con el servicio web actualizando la información publicada por el administrador de la información y los web service, todo en tiempo real..

Navegador web

Empleo en tu bolsillo

Servicio de empleo movil

Empleo en tu bolsillo

Empleo en tu bolsillo

Empleo en tu bolsillo

Empleo en tu bolsillo

Google Play

Dispositivo Android

Servidor

Dispositivo iOS

App Store

Ilustración 4 - Vista de despliegue

* Google Play: es el mecanismo ofrecido por Google para la distribución de aplicaciones en la plataforma Android, mediante el cual los usuarios ingresan a la tienda en línea y descargan las aplicaciones. En este sitio se ubicarán los archivos de empaquetamiento de la aplicación con formato APK (Android Package Archive) para Android correspondiente al módulo móvil para la comisión nacional del servicio civil.

Se deberá contar con una cuenta para la distribución de las aplicaciones en esta tienda.

* App Store: es el mecanismo ofrecido por Apple para la distribución de aplicaciones en la plataforma iOS, mediante el cual los usuarios ingresan a la tienda en línea y descargan las aplicaciones. En este sitio se ubicarán los archivos empaquetados de las aplicaciones con formato IPA (iPhone Application) para iOS correspondiente al módulo móvil para la comisión nacional del servicio civil.
* Se deberá contar con una cuenta para la distribución de las aplicaciones en esta tienda.
* Servicios empleo móvil: Corresponde al servidor que ofrece el servicio web para obtener y enviar información ya sea desde la aplicación móvil o desde la página web. La disponibilidad, desempeño y concurrencia de los servicios son responsabilidad de la entidad.
* Dispositivos Móviles Android: corresponde a cualquier dispositivo móvil de un ciudadano con sistema operativo Android donde se realiza la instalación de las soluciones móviles. La versión del Android debe ser 3.0 o superior.
* Dispositivos Móviles iOS: corresponde a cualquier dispositivo de un ciudadano con sistema operativo iOS donde se realiza la instalación de las soluciones móviles. La versión del iOS debe ser 5.0 o superior.
* Navegador web: corresponde a un cliente web sea Chrome o Firefox, que permite la consulta de sitios web cuando se dispone de una conexión activa.

15. DI - VISTA DE INTEGRACIÓN CON SISTEMAS EXTERNOS

La aplicación móvil se comunica con el webservice definido por la entidad por medio de comunicación segura, la aplicación utiliza además enlaces externos a

redes sociales para que los usuarios puedan compartir información y utiliza el

servicio de externo de Google Maps para diferentes módulos de la aplicación



Ilustración 7. Vista de Integración con sistemas externos.

16. DI - VISTA DE PARAMETRIZACIÓN DEL SISTEMA

L

a vista que cumple con los parámetros del sistema son aquellos aspectos concernientes a valores y parámetros sujetos a cambios, que inciden en la aplicación móvil.

Debido a la naturaleza stand alone de las aplicaciones móviles, todos los cambios relacionados con valores y parámetros dentro de la aplicación implican la realización de una nueva compilación, y la generación de una nueva versión de la misma por cada plataforma, la cual podrá ser distribuida como actualización en las respectivas tiendas de aplicaciones.

Los siguientes serán los aspectos parametrizables de la aplicación:

**a)** Retorna el listado de vacantes según los siguientes parametros:

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| palabra | Palabra clave a buscar | Texto |
| tipo | Tipo de oportunidad | Entero |
| Salario | Rango de salario | Entero |
| Experiencia | Tipo de experiencia | Entero |
| Nivel | Nivel educativo | Entero |
| Municipio | Municipio a consultar | Entero |

**b)** Un método que permite consultar las vacantes en un mapa según estos parametros:

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| palabra | Palabra clave a buscar | Texto |
| tipo | Tipo de oportunidad | Entero |
| Salario | Rango de salario | Entero |
| Experiencia | Tipo de experiencia | Entero |
| Nivel | Nivel educativo | Entero |
| Municipio | Municipio a consultar | Entero |

**c)** Se requiere un método que retorna el listado de las vacantes del empleador.

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| Empleador | Empleador que desea consultar sus vacantes | Texto |

**d)** Se requiere un método que permita registrar una vacante

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| ID | null | Texto |
| Titulo | Titulo de la vacante | Texto |
| TipoID | Rango de salario | Entero |
| Descripcion | Tipo de experiencia | Entero |
| NumVacantes | Nivel educativo | Entero |
| Cargo | Municipio a consultar | Entero |
| Salario |  | Entero |
| Sector |  | String |
| ExperienciaID | Tipo de experiencia | Entero |
| Nivel\_EstudiosID | Nivel de estudios requerido | Entero |
| Profesion |  | String |
| Municipio |  | Entero |
| Departamento |  | Entero |
| Fecha\_publicacion |  | Fecha |
| Fecha\_vencimiento |  | Fecha |
| Latitud |  | String |
| Longitud |  | String |
| Empleador |  | String |
| Telefono |  | String |

**e)** Se requiere un método que permita consultar una vacante en específico

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| vacanteID | Id de la vacante a consultar | Entero |

**f)** Método que permita modificar una vacante

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| ID | Id de la vacante a modificar | Texto |
| Titulo | Titulo de la vacante | Texto |
| TipoID | Rango de salario | Entero |
| Descripcion | Tipo de experiencia | Entero |
| NumVacantes | Nivel educativo | Entero |
| Cargo | Municipio a consultar | Entero |
| Salario |  | Entero |
| Sector |  | String |
| ExperienciaID | Tipo de experiencia | Entero |
| Nivel\_EstudiosID | Nivel de estudios requerido | Entero |
| Profesion |  | String |
| Municipio |  | Entero |
| Departamento |  | Entero |
| Fecha\_publicacion |  | Fecha |
| Fecha\_vencimiento |  | Fecha |
| Latitud |  | String |
| Longitud |  | String |
| Empleador |  | String |
| Telefono |  | String |

**g)** Se requiere un método que permita eliminar una vacante en específico

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| **vacanteID** | **Id de la vacante a consultar** | **Entero** |

**h)** Cambiar estado de una vacante

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| VacanteID | Id de la vacante a modificar | Entero |
| Empleador | Empleador propietario de la vacante | String |
| Estado |  |  |

1. Consultar preguntas frecuentes.

Este método no requiere de parametros.

**j)** Consultar denuncias de una vacante.

|  |  |  |
| --- | --- | --- |
| **Nombre del parámetro** | **Descripción** | **Tipo de dato** |
| VacanteID | Id de la vacante a consultar las denuncias | Entero |

* 1. **ESPECIFICACIÓN DEL SERVICIO WEB**

El protocolo de comunicación que se utiliza desde el web service es SOAP por HTTP, con consumo de datos xml, la aplicación no funciona si no tiene acceso a internet.

* 1. **INTEGRACIÓN CON REDES SOCIALES**

La aplicación tiene interacción con redes sociales, específicamente Facebook y twitter para lo cual el usuario tendrá la opción de compartir un mensaje precargado en su perfil que podrá modificar, debe iniciar sesión con su respectiva cuenta.

17. DI - HISTORIAS DE USUARIO

* 1. Oportunidades de empleo

|  |  |
| --- | --- |
| Oportunidades de empleo | |
| NHU: 1. | **Usuario:** Usuario de la aplicación móvil |
| Descripción:  El usuario ingresa a la aplicación móvil desde su Smartphone, la aplicación presenta la pantalla inicial que permite realizar una búsqueda inmediata de las ofertas de empleo.  El usuario tiene diferentes filtros para aplicar a su búsqueda además de consultarlas graficadas en el mapa del territorio nacional. | |
| Cómo Probar   * Como usuario ingresar a la aplicación y verificar la existencia de la opción “Oportunidades de empleo”. * Aplicar filtros y buscar, verificar la existencia de mensajes y/o ofertas * Presionar el botón “Ver vacantes en el mapa”. * Verificar que se visualizan puntos en el mapa y seleccionar uno de estos y luego el botón “Ver detalles” * Verificar que aparece el detalle de la oferta. | |

* 1. MIS VACANTES FAVORITAS

|  |  |
| --- | --- |
| Mis vacantes favoritos | |
| NHU: 2. | **Usuario:** Usuario de la aplicación móvil |
| Descripción:  El usuario ingresa a la aplicación y selecciona la opción “Mis vacantes favoritos”, en esta encontrara el listado de las ofertas marcadas como favoritas | |
| Cómo Probar:   * Como usuario ingresar a la aplicación y verificar la existencia de la opción “Mis vacantes favoritos”. * Verificar que toda la información descrita se visualiza correctamente. | |

* 1. Sección del empleador

|  |  |
| --- | --- |
| Sección del empleador | |
| NHU: 3 | **Usuario:** Usuario de la aplicación móvil |
| Descripción:  El usuario ingresa a la aplicación y en el menú selecciona la opción “Sección del empleador”  Una vez el usuario haya seleccionado esta opción puede iniciar sesión para consultar su información privada y única. | |
| Cómo Probar:  - Ingresar a la opción indicada e iniciar sesión con datos correctos y registrados. | |

* 1. PREGUNTAS FRECUENTES

|  |  |
| --- | --- |
| PREGUNTAS FRECUENTES | |
| NHU: 7 | **Usuario:** Usuario de la aplicación móvil |
| Descripción:  El usuario ingresa a la aplicación y en el menú selecciona la opción “Preguntas frecuentes”.  Una vez el usuario haya seleccionado esta opción puede consultar la información actualizada que puede ser útil.  El contenido de la página debe incluir:   * Preguntas * Respuestas respectivas | |
| Cómo Probar:  - Como usuario consultar la opción “Preguntas frecuentes” del menú principal, verificar que la aplicación despliega la información correspondiente. | |

* 1. ACERCA DE

|  |  |
| --- | --- |
| ACERCA DE | |
| NHU: 7 | **Usuario:** Usuario de la aplicación móvil |
| Descripción:  El usuario ingresa a la aplicación y en el menú selecciona la opción “Acerca de”.  Una vez el usuario haya seleccionado esta opción puede consultar la información actualizada del “acerca de la aplicación” brindada por la entidad.  El contenido de la página debe incluir:  - Versión de la aplicación  - Equipo de desarrollo  - Mensaje importante  - Entidades de apoyo | |
| Cómo Probar:  - Como usuario consultar la opción “Acerca de” del menú principal, verificar que la aplicación despliega la información correspondiente. | |

ESTIMACIÓN DE ESFUERZO

18. EE - ESTIMACIÓN DE ESFUERZO

C

omo resultado de la estimación de esfuerzo se identificó que la aplicación Servicio de empleo móvil cuenta con menos de 600 puntos de función, por lo cual se considera viable su inclusión dentro de la iteración.

Para llegar a ésta conclusión se tuvieron en cuenta las siguientes características:

1. Comunicación de datos.
2. Procesamiento de datos distribuidos.
3. Desempeño
4. Configuración altamente usada.
5. Tasa de la transacción.
6. Entrada de datos en línea.
7. Eficiencia usuario final.
8. Actualización en línea.
9. Procesamiento complejo.
10. Reusabilidad.
11. Facilidad de instalación.
12. Facilidad de uso.
13. Portabilidad.
14. Facilidad de cambio.

Y se consideraron como factores de peso para la creación de la aplicación la complejidad de cada historia de usuario y las siguientes categorías:

1. Archivo Interno Lógico
2. Archivo de Interface Externa
3. Entrada Externa
4. Salida Externa
5. Consulta Externa

PLAN DE PRUEBAS DE ACEPTACIÓN

19. PPA - PLAN DE PRUEBAS DETALLADO

**19.1 METODOLOGÍA DE PRUEBAS**

A continuación se presenta la metodología de pruebas que se va realizar a la Aplicación Servicio de empleo móvil en cada uno de los siguientes ambientes: Ambiente de desarrollo, Ambiente de pre-producción, Ambiente de producción y Ambiente de estabilización.

**19.2 AMBIENTE DE DESARROLLO**

En este ambiente, las pruebas son realizadas por el equipo de desarrolladores al finalizar la construcción de la Aplicación. Los desarrolladores realizarán las pruebas unitarias con base en los casos de prueba diseñados para la aplicación que pueden ser consultados en el documento de la aplicación y que sigue los lineamientos del diseño allí descrito.

**19.3 AMBIENTE DE PRUEBAS PRE-PRODUCCIÓN**

En este ambiente se realizarán las pruebas internas por parte del Consorcio Software 2012. Los tipos de pruebas a ejecutar son:

* Pruebas de funcionalidad
* Pruebas de interfaz de usuario
* Pruebas de bases de datos e integridad de datos
* Pruebas de desempeño
* Pruebas de estrés
* Pruebas de carga
* Pruebas de recuperación a fallas

La ejecución de estas pruebas, las realizarán los analistas de aseguramiento de calidad del equipo de trabajo del Consorcio Software 2012 con base en los casos de prueba diseñados para la aplicación.

Los ciclos de prueba previstos para ejecutarse en este ambiente son:

* Primer ciclo con su respectivo re-test.
* Segundo ciclo con su respectivo re-test.
* Ciclo de regresión para las historias de usuario que lo ameriten.

El objetivo de los ciclos uno (1) y dos (2) es ejecutar todos los casos de prueba que se tengan definidos y efectuar su correspondiente retest, es decir, luego de solucionados los bugs que fueron reportados a través de la herramienta Jira, se procederá a verificarlos y se cerrarán si cumplen con el caso de prueba satisfactoriamente.

Después de finalizar los ciclos uno y dos, se procederá a realizar el ciclo de regresión, cuyo objetivo es ejecutar nuevamente los casos de prueba que ya se habían aplicado a las historias para asegurarse que los cambios no han propagado efectos colaterales adversos.

Los fallos encontrados serán reportados por medio de la herramienta de Google Drive:

<http://bog.ospinternational.com:8889/secure/Dashboard.jspa>

**19.4 AMBIENTE DE PRUEBAS PRODUCCIÓN**

En este ambiente se realizarán las pruebas por parte de los consultores de Gobierno en Línea y el Servicio de empleo, como parte del proceso de Aseguramiento de la calidad, estas pruebas se realizarán con base en los casos de prueba diseñados para la aplicación.

En el ambiente de producción se podrán realizar pruebas que no estén especificadas en el documento de criterios de aceptación, lo anterior teniendo en cuenta que se pueden encontrar fallos a nivel de usuario que no fueron tenidos en cuenta en los casos de pruebas.

Los fallos encontrados serán reportados por medio de la herramienta de seguimiento Google Drive.

<https://docs.google.com/a/ospinternational.com/spreadsheet/ccc?key=0AlZIlytkJ-T6dDRkalZvRnY4MGo3X1A4dUZyVlE1RWc&usp=drive_web#gid=12>

**19.5 AMBIENTE DE ESTABILIZACIÓN**

Este ambiente corresponde a subir las soluciones a las tiendas de aplicaciones Google Play y Apps Store y publicar el sitio web en un servido destinado para eso. Las cuentas de desarrollo a través de las cuales éstas se publicarán, serán provistas por el Consorcio Software 2012 y el servicio de emplo será el responsable de su mantenimiento y operación.

El proceso de aprobación en cada una de las tiendas estará a cargo del Consorcio Software 2012.

La aplicación deberá entregarse en funcionamiento por medio de un acta entre el Consorcio y la entidad, de acuerdo con los criterios de aceptación previamente definidos, incorporando toda la funcionalidad. Posteriormente, se realizará el apoyo a la operación de la solución por cuatro semanas, donde se deben realizar los ajustes que se identifiquen y que sean necesarios para la correcta operación de la misma.

20. PPA - TIPOS DE PRUEBA

**20.1 PRUEBAS DE FUNCIONALIDAD**

El objetivo de estas pruebas es asegurar la funcionalidad establecida en la definición de alcance, incluyendo la navegación, entrada de datos y su procesamiento.

Ejecutar cada caso de prueba para verificar que se obtienen los resultados esperados, que se muestran los mensajes de error o advertencia adecuados.

**20.2 PRUEBAS DE INTERFAZ DE USUARIO**

Las pruebas de interfaz de usuario están orientadas hacia la funcionalidad y usabilidad. El método utilizado será por inspección, revisando el diseño y la interfaz de forma sencilla e intuitiva, así como la interacción con la aplicación de manera natural y lógica.

Las pruebas están orientadas a la navegación a través de los objetos de prueba reflejando apropiadamente las funcionalidades, incluyendo los saltos entre campos y la utilización de distintos métodos de acceso (tabulador, zoom y desplazamientos, doble clic (tap)).

Se deben verificar los objetos y características de las ventanas, tales como menús, tamaño, posición, estado y foco se comportan según los acuerdos establecidos en la definición de alcance.

Se medirá el nivel de usabilidad a través de la interacción táctil de la aplicación, verificando la acción de tocar, arrastrar, presionar, extender, girar.

**20.3 PRUEBAS DE BASES DE DATOS E INTEGRIDAD DE DATOS**

Estas pruebas están orientadas a la verificación del acceso a la base de datos y de las respuestas de las consultas, sin pérdida o corrupción de datos.

Para la Integridad de datos se verificará si la información presentada corresponde a la temática escogida.

**20.4 PRUEBAS DE DESEMPEÑO**

Los tiempos de respuesta y funcionamiento de las aplicaciones que utilicen Servicios Web dependerán de la disponibilidad y desempeño tanto del esquema de conexión del móvil (4G, 3G, HSDPA, WiFi) como del servicio mismo ofrecido por la entidad respectiva.

Verificar la estabilidad de la interfaz, la interfaz debe conservar fluidez, estabilidad y tiempo de respuesta favorables para la interacción con el usuario.

**20.5 PRUEBAS DE ESTRÉS**

No aplica. No se va a verificar el límite del sistema así como su comportamiento de acuerdo a una concurrencia ya que serán propias del cliente.

**20.6 PRUEBAS DE CARGA**

No aplica. No se va a verificar el límite del sistema así como su comportamiento de acuerdo a una concurrencia ya que serán propias del cliente.

**20.7 PRUEBAS DE RECUPERACIÓN DE FALLAS**

Se trata de conseguir que el sistema responda ante la falta de conexión a internet por medio de mensajes, y que, en caso de que se restablezca la conexión, pueda seguir operando con normalidad.

1. PPA - AMBIENTE DE PRUEBAS

El ambiente de pruebas que se indica a continuación corresponde con lo requerido para los ambientes de desarrollo y preproducción.

* 1. **AMBIENTE DE DESARROLLO**
  2. **HARDWARE PARA PRUEBAS EN AMBIENTE DE DESARROLLO**

En la siguiente tabla se indica el hardware requerido para las pruebas en el ambiente de desarrollo.

Tabla 1. Hardware requisito para ambiente de desarrollo

| DISPOSITIVO | CANTIDAD | DESCRIPCIÓN |
| --- | --- | --- |
| Tablet | 2 | iPad Mini, Tablet con sistema operativo Android |
| Teléfono | 4 | iPhone 3, iPhone 4S, 1 celular con sistema operativo Android 2.3.2, 1 celular con sistema operativo Android 4.3. |
| Emulador | 2 | Xcode, Emulador de Android |
| Computador con clientes web y acceso a internet. | 2 | Diferentes sistemas operativos y navegadores (Chrome y Firefox) |

* 1. **SOFTWARE**

En la siguiente tabla, se presenta el software requerido para la configuración de las pruebas en ambiente de desarrollo

Tabla 2. Software requisito para ambiente de desarrollo

| RECURSO | CANTIDAD | DESCRIPCIÓN |
| --- | --- | --- |
| Android | 3 | Versión 4.2. del sistema operativo Android en adelante con nivel de api 17 revisión en emuladores de pruebas LG NEXUS ONE, LG NEXUS 4 Y TABLET LG NEXUS 7 SAMSUNG GALAXY S, S2, S3 Y S4  Teléfono de pruebas Samsung Galaxy S4.  Versión 2.3.6 del sistema operativo Android en teléfono de pruebas Samsung Galaxy Y |
| Ios | 5 | Versión 5.1 en adelante en dispositivos IOS teléfonos de pruebas iPhone 4S, iPhone 5, iPad 2, iPad 3, iPad Mini, iPod Touch 4G |
| Sistemas operativos y navegadores | 2 | Windows y Mac  Chrome y Firefox |

**21.4 AMBIENTE DE PREPRODUCCIÓN**

**21.5 HARDWARE PARA PRUEBAS EN AMBIENTE DE PREPRODUCCIÓN**

En la siguiente tabla se indica el hardware requerido para las pruebas en el ambiente de preproducción.

Tabla 3. Hardware requisito para ambiente de preproducción

| DISPOSITIVO | CANTIDAD | DESCRIPCIÓN |
| --- | --- | --- |
| Tableta | 3 | * Dispositivo móvil tipo tablet con sistema operativo Android, resolución de pantalla 7” WSVGA TFT LCD (1024\*600). * Dispositivo móvil tipo tablet con sistema operativo iOS. Resolución de pantalla 7,9” (1024 \* 768) – (163 pixeles por pulgada). * Dispositivo móvil iPod Touch 4G con sistema operativo iOS Resolución de pantalla 3,5” (960 \* 640) – (326 pixeles por pulgada) |
| Teléfonos | 7 | * Android Motorola RZR HD * Samsung Galaxy Y * Samsung Galaxy Young 2 * Samsung Galaxy Note * Samsung Galaxy S3 Mini * Iphone 3 * Iphone 4S |

**21.6 SOFTWARE PARA PRUEBAS EN AMBIENTE DE PREPRODUCCIÓN**

En la siguiente tabla, se presenta el software requerido para la configuración de las pruebas en ambiente de preproducción

Tabla 4. Software requisito para ambiente de preproducción

| RECURSO | CANTIDAD | DESCRIPCIÓN |
| --- | --- | --- |
| Android | 4 | versión 4.1.2  versión 4.0.3  versión 4.0.4  versión 2.3.6 |
| IOS | 2 | versión 6.1.5  versión 7.0.4 |

Criterios de aceptación

1. CA - CRITERIOS DE ACEPTACIÓN

E

stablece los criterios de aceptación de la Aplicación Servicio de empleo móvil desarrollada la cual contiene los casos de prueba diseñados de acuerdo a los diferentes escenarios que deben probarse para la aplicación.

El propósito de los Criterios de Aceptación es definir las pruebas a realizar con el fin de validar el correcto funcionamiento de la aplicación. Los tipos de prueba a realizar y que se encuentran descritos en el Plan de Pruebas de aceptación son:

* Pruebas Funcionales
* Pruebas de Interfaz de Usuario
* Pruebas de Base de datos e integridad de datos
* Pruebas de Desempeño
* Pruebas de Recuperación de fallas

Para aceptar el software hecho por el grupo desarrollador del reto Servicio de empleo, se requiere que éste cumpla las siguientes condiciones:

* Aplicación desarrollada para las plataformas iOS y Android.
* Sitio web disponible desde cualquier lugar con acceso a internet.
* Pruebas técnicas y funcionales satisfactorias realizadas al software por parte del Operador las cuales estarán reflejadas en el informe de pruebas.
* Las pruebas que se realizarán a la aplicación están establecidas en la sección Plan de Pruebas Detallado.

A continuación se describen los casos de prueba que debe aprobar la aplicación de manera satisfactoria para que sea aceptada por el Consorcio Software 2012 y el Ministerio de Tecnologías de la Información y las Comunicaciones.

1. CA - CASOS DE PRUEBA DE LA APLICACIÓN
   1. **CASOS DE PRUEBA FUNCIONALES**

Los casos de prueba funcionales están orientados a la funcionalidad de las aplicaciones en los ambientes iOS y Android y del sitio web.

Los casos de prueba se encuentran indicados en las historias de usuario.

* 1. **CASOS** **DE PRUEBA DE INTERFAZ DE USUARIO**

Los casos de prueba de Interfaz de usuario están orientados a las funciones de los dispositivos móviles.

|  |  |
| --- | --- |
| **Nº Caso prueba** |  |
| **Historia de Usuario Asociada** | N.A. |
| **Propósito de la prueba:** | Verificar la usabilidad de la aplicación (Tocar) |
| **Descripción del caso de prueba:** | Verificar la respuesta de la aplicación a la acción de tocar sobre la pantalla. |
| **Datos de Entrada:** | NA |
| **Resultado Esperado:** | La aplicación responde a las acciones de usabilidad. |
| **Pre-condiciones:** | |
| * El usuario ha instalado y abierto la aplicación. | |
| **FLUJO BÁSICO (Tocar )** | |
| 1. El usuario verifica las opciones de Tocar. 2. \*El sistema responde a la acción realizada tocar. 3. Muestra la información solicitada. | |
| **FLUJO ALTERNO 1 ( Menús) 2\*** | |
| 1. \*El sistema no responde a la acción tocar. 2. Fin del caso de prueba. | |
| **Post-condiciones:** | |
| * El sistema permite mostrar las opciones cuando se toca la pantalla. | |

|  |  |
| --- | --- |
| **Nº Caso prueba** |  |
| **Historia de Usuario Asociada** | N.A. |
| **Propósito de la prueba:** | Verificar la usabilidad de la aplicación (Girar) |
| **Descripción del caso de prueba:** | Verificar la respuesta de la aplicación a la acción de girar el dispositivo. |
| **Datos de Entrada:** | NA |
| **Resultado Esperado:** | La aplicación responde a las acciones de usabilidad. |
| **Pre-condiciones:** | |
| * El usuario ha instalado y abierto la aplicación. | |
| **FLUJO BÁSICO (Girar)** | |
| 1. El usuario verifica la opción girar, es decir gira el dispositivo móvil. 2. \*El sistema responde a la acción realizada girar. 3. Muestra la información solicitada. | |
| **FLUJO ALTERNO 1 ( Variables) 2\*** | |
| 1. \*El sistema no responde a la acción girar. 2. Fin del caso de prueba. | |
| **Post-condiciones:** | |
| * El sistema permite mostrar las opciones cuando se pellizca la pantalla. | |

* 1. **CASOS DE PRUEBA ESTRÉS**

No se va a verificar el límite del sistema así como su comportamiento de acuerdo a una concurrencia ya que serán propias del cliente, por lo tanto estas pruebas no aplican

* 1. **CASOS DE PRUEBA DE CARGA**

No se va a verificar el límite del sistema así como su comportamiento de acuerdo a una concurrencia ya que serán propias del cliente, por lo tanto estas pruebas no aplican.

* 1. **CASOS DE PRUEBA DE BASES DE DATOS E INTEGRIDAD DE DATOS**

|  |  |
| --- | --- |
| **Nº Caso prueba** |  |
| **Historia de Usuario Asociada** | N.A. |
| **Propósito de la prueba:** | Verificar la integridad de datos de la aplicación. |
| **Descripción del caso de prueba:** | Verificar que la información visualizada corresponda con la información relacionada con el set de datos. |
| **Datos de Entrada:** | NA |
| **Resultado Esperado:** | La aplicación muestra la información correctamente. |
| **Pre-condiciones:** | |
| * El usuario ha instalado y abierto la aplicación. | |
| **FLUJO BÁSICO ( )** | |
| 1. El usuario Filtra la información a consultar. 2. La aplicación presenta la información correspondiente al filtro de la consulta. 3. El usuario verifica que visualmente se refleje la información consultada. | |
| **Post-condiciones:** | |
| * El sistema muestra la información correctamente. | |

* 1. **PRUEBAS DE DESEMPEÑO**

|  |  |
| --- | --- |
| **Nº Caso prueba** |  |
| **Historia de Usuario asociada:** | NA |
| **Propósito de la prueba:** | Confirmar que la aplicación se puede instalar en diferentes dispositivos móviles con sistema operativo Android o iOS y que el sitio web es accesible desde diferentes puntos con acceso a internet. |
| **Descripción del caso de prueba:** | Se instala la aplicación en un dispositivo móvil y se verifica que instale correctamente y que se pueda abrir. |
| **Datos de Entrada:** | NA |
| **Resultado Esperado:** | El descargue de la aplicación es exitoso. La aplicación instala correctamente en el dispositivo y se puede abrir. |
| **Pre-condiciones:** | |
| * La aplicación deberá estar disponible en Testflight. * El usuario deberá estar autorizado en Testflight. * Para los dispositivos con sistema operativo iOS, el usuario deberá haber reportado el UDID al administrador de Testflight. | |
| **FLUJO BÀSICO (Instalar la aplicación en un dispositivo móvil)** | |
| 1. El usuario ingresa y se autentica en Testflight. 2. El usuario selecciona la última versión de la aplicación compatible con el sistema operativo de su dispositivo móvil. 3. El sistema instala correctamente la aplicación en el dispositivo. 4. El sistema muestra correctamente el ícono representativo de la aplicación. 5. El usuario selecciona y abre la aplicación con un tap sobre el icono. 6. El sistema ejecuta la aplicación correctamente. | |
| **Pos-condiciones:** | |
| * La aplicación se ejecuta correctamente en los dispositivos móviles con sistema operativo Android y/o iOS. | |

1. IP - HISTORIAS DE USUARIO PROBADAS

A continuación se presenta el listado de las historias de usuario probadas en ambiente de preproducción de la Aplicación Móvil para la consulta de oportunidades de empleo en todo el territorio nacional.

Los tipos de pruebas que se aplicaron se encuentran: Pruebas de funcionalidad,

Pruebas de interfaz de usuario, Pruebas de Bases de Datos e Integridad de Datos,

Pruebas de Desempeño y Pruebas de Recuperación de Fallas.

Tabla 5. Historias de Usuario

|  |  |
| --- | --- |
| **Número De La Historia De Usuario** | **Nombre Historia De Usuario** |
| 1 | Oportunidades de empleo |
| 2 | Mis vacantes favoritos |
| 3 | Sección del empleador |
| 4 | Preguntas frecuentes |
| 5 | Acerca de |

* 1. **CASOS DE PRUEBA DE RECUPERACIÓN DE FALLAS**

|  |  |
| --- | --- |
| **Nº Caso prueba** |  |
| **Historia de Usuario Asociada** | N.A. |
| **Propósito de la prueba:** | Validar que la aplicación continúa funcionando después de una caída en el servicio de datos. |
| **Descripción del caso de prueba:** | Verificar el manejo de los mensajes informativos en la aplicación. Cuando existan caídas inesperadas del servicio, la aplicación debe manejar la excepción y advertir con mensajes claros al usuario. |
| **Datos de Entrada:** | NA |
| **Resultado Esperado:** | La aplicación maneja mensajes informativos |
| **Pre-condiciones:** | |
| * La aplicación ha sido instalada correctamente en el dispositivo móvil. | |
| **FLUJO BÁSICO ( )** | |
| 1. El usuario inicia por primera vez la aplicación sin tener conexión de datos 2. La aplicación muestra el mensaje alusivo a la falta de conexión a internet para acceder a las funcionalidades. | |
| **FLUJO ALTERNO** | |
| 1. El usuario selecciona una a una las funcionalidades. 2. El usuario inhabilita el servicio de acceso a datos del dispositivo móvil (WiFi o Conexión de datos 3G). 3. La aplicación presenta el mensaje que indica que se ha presentado la perdida de conexión a internet. | |
| **Post-condiciones:** | |
| * La aplicación muestra mensajes informativos. | |

INFORME DE PRUEBAS

1. IP - REGISTRO Y CLASIFICACIÓN DE INCIDENCIAS

El proceso registro de las incidencias debe hacerse a través de la herramienta de gestión JIRA. Cuando se identifica un hallazgo, se debe suministrar la

Información básica necesaria para que el desarrollador pueda hacer el

Procesamiento del incidente (hora, descripción del incidente, sistemas afectados,

etc).Para registrar una incidencia en JIRA se ingresan como mínimo los siguientes datos:

* **Proyecto:** se identifica el proyecto al cual pertenece la aplicación. Para éste caso corresponde a Agro Insumos Colombia.
* **Tipo de incidencia:** dado que JIRA permite el seguimiento de diferentes tipos de incidencia, los errores de las aplicaciones se registran como Defecto que corresponde a un problema que dificulta o impide el funcionamiento del producto.
* **Sumario:** corresponde a un breve resumen que permita identificar la incidencia.
* **Prioridad:** una incidencia tiene un nivel de prioridad que indica su importancia. Puede ser bloqueadora, crítica, mayor, menor, trivial.
* **Responsable:** se asigna al líder del equipo desarrollador. Para este caso, Cristian Camilo Montes
* **Informante:** para realizar el seguimiento a las incidencias se asigna como informante al usuario que hace el registro.
* **Descripción:** se detalla el error presentado en la aplicación y la forma como se presentó con el fin de que el desarrollador lo pueda reproducir.
* **Adjunto:** de ser necesario se puede cargar una o varias imágenes del defecto encontrado.

1. IP - GESTIÓN DE INCIDENCIAS
   1. **ANÁLISIS Y REPORTE DE INCIDENCIAS**

Con el fin de completar el proceso de aseguramiento de la calidad, se ejecutaron los casos de pruebas diseñados para la aplicación servicio de empleo móvil , basados en los criterios de aceptación del presente documento

La ejecución de estas pruebas la realizaron los analistas de aseguramiento de calidad del equipo de trabajo del Consorcio Software 2012, los miembros del equipo de trabajo de la Dirección de Gobierno en Línea y Ministerio de Justicia y del Derecho, con base en las historias de usuario definidas en el presente documento.

Los ciclos de prueba que se ejecutaron en el ambiente de preproducción fueron los siguientes:

* Primer ciclo con su respectivo re-test.
* Segundo ciclo con su respectivo re-test.
* Ciclo de regresión para las historias de usuario que lo ameritan.
  1. **FLUJO DE TRABAJO PARA RESOLUCIÓN DE INCIDENCIAS**

La herramienta JIRA se encarga de enviar una notificación automática al responsable de la incidencia para que éste inicie el proceso de resolución.

El desarrollador debe dar respuesta a través de la herramienta, modificando el estado de la incidencia según el flujo de trabajo establecido.

Cada incidencia tiene un estado que indica la etapa en que se encuentra. En el flujo de trabajo por defecto, las incidencias empiezan estando Abiertas, pasan a estar En Progreso, Resueltas y finalmente Cerradas. Otros flujos de trabajo pueden tener otras transiciones de estado.

* 1. **CIERRE DE INCIDENCIAS**

Una incidencia puede resolverse de diversas maneras, siendo sólo una de ellas Solucionarlo. Las resoluciones definidas son:

* **Solucionada**: Se ha desarrollado y probado una solución para esta incidencia.
* **No se arreglará:** El problema descrito en la incidencia no se solucionará.
* **Duplicado:** La incidencia es un duplicado de una incidencia existente.
* **Incompleta:** El problema no está completamente descrito.
* **No Reproducible:** Todos los intentos de reproducción de esta incidencia fueron fútiles, o no había suficiente información disponible para reproducir el problema reportado. La lectura del código fuente no ha producido ninguna pista de por qué este comportamiento ocurriría. Si aparece más información en el futuro, por favor vuelva a abrir la incidencia.
* **No es un Error:** Aplica para las incidencias que no son realmente errores de la aplicación.

En el proceso de aseguramiento de calidad de la Aplicación DROOG, todas las incidencias están siendo resueltas por el desarrollador.

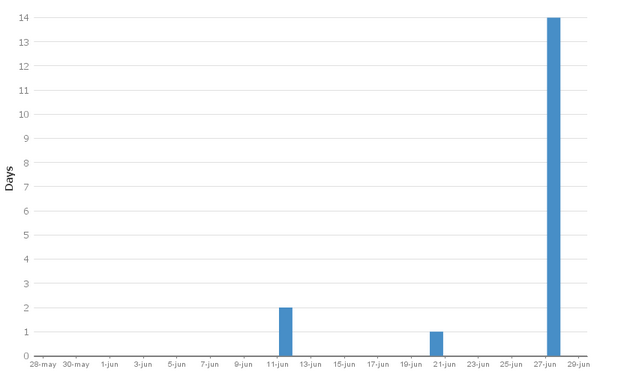


Ilustración 8. Tiempo de Resolución.

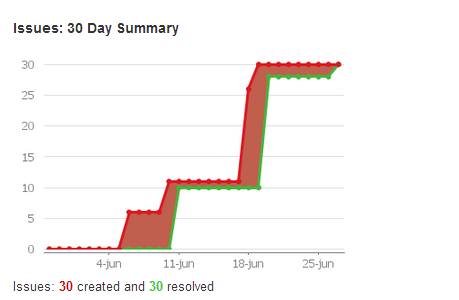


Ilustración 9.Incidencias creadas vs. Resueltas

MANUAL DE INSTALACIÓN Y CONFIGURACIÓN

1. MI - PRE-REQUISITOS
   1. **PRE-REQUISITOS HTML5 – CSS3 – JAVASCRIPT – jQUERY**

Para configurar el ambiente de desarrollo para HTML5 – CSS3 y Javascript no se requieren prerrequisitos especiales. Cómo prerrequisitos mínimos se tienen:

Tabla 7. Pre-Requisitos

|  |  |
| --- | --- |
| **Característica** | **Definición** |
| **Dispositivo** | * Computador personal (PC) |
| **Requerimientos de Máquina** | * Windows:   + Windows XP o Superior.   + Ubuntu 10.04 LTS o Superior   + Mac OS 10.7 o superior * Espacio de almacenamiento:   + 2 GB. * Memoria RAM:   + 1GB |
| **Repositorio** | * Debe existir una cuenta en el repositorio online GitHub: <https://github.com/>. |
| **Compilador** | * Se utiliza la herramienta xcode y eclipse con proyectos generados a partir de phonegap |

**27.2 AMBIENTE DE PRUEBAS**

Tabla 8. Ambiente de pruebas

|  |  |
| --- | --- |
| **Característica** | **Definición** |
| **Dispositivos** | * Tablet. * Teléfono Móvil. |
| **Sistema Operativo:** | * Android:   + Android 2.3 o Superior. * iOS:   + iOS 5.0 o superior. |
| **Software** | * Para las aplicaciones de iOS se requiere tener instado iTunes en un computador personal. |
| **Ambiente de Pruebas** | * Se debe contar con una cuenta en Test Flight. * Se debe contar con una cuenta de Apple Developer activa. |

* 1. AMBIENTE DE PRODUCCIÓN

Tabla 9. Ambiente de Producción

|  |  |
| --- | --- |
| **Característica** | **Definición** |
| **Dispositivos** | * Tablet * Teléfono Móvil. |
| **Sistema Operativo:** | * Android:   + Android 2.3 o Superior. * iOS:   + iOS 5.0 o superior. |
| **Espacio Disponible** | 6Mb |

* 1. **PRE-REQUISITOS ADICIONALES**

Tabla 10. Pre-requisitos adicionales

|  |  |
| --- | --- |
| **Característica** | **Definición** |
| **Acceso a Internet** | El dispositivo móvil debe contar con un plan de datos para poder conectarse a internet o en su defecto tener acceso a una red wifi que le permita conectarse a Internet. |
| **Entrada Táctil** | El dispositivo móvil debe tener activada y funcionando su interfaz táctil. |
| **Disponibilidad de servicios web.** | El servicio web que consume la aplicación, debe mantenerse activo permanentemente. |

1. MI - PREPARACIÓN DE ADMINISTRADORES DE LA APLICACIÓN Y ARCHIVOS DE INSTALACIÓN
   1. **PREPAPARACIÓN DE ADMINISTRADORES DE LA APLICACIÓN**

El consorcio realizará una capacitación técnica de las aplicaciones móviles al personal que la entidad asigne, con el objetivo que la entidad administre de manera propia la aplicación.

* 1. **AMBIENTE DE DESARROLLO**

Esta aplicación se desarrolló en lenguaje HTML5, Css y javascript; por tanto, se requiere de un editor de HTML. Inicialmente la aplicación se desarrolla como si se tratara de un sitio web, utilizando un archivo en formato html denominado index, el cual hace el papel de página principal o home de la aplicación.

* 1. **ESTRUCTURA DE CARPETAS**

La estructura de carpetas del sitio para el desarrollo de la aplicación se describe a continuación:

El desarrollo se encuentra publicado en GitHub en la siguiente dirección: https://github.com/cristian4063/EmpleoMovil

Dentro de la subcarpeta code se encuentran alojadas las siguientes carpetas:

* **styles:** En la carpeta styles se encuentran guardados los archivos de estilos css que utiliza la aplicación
* **images:** En esta carpeta se almacenan todas las imágenes que se utilizan en las interfaces y el contenido de la aplicación.
* **scripts:** En esta carpeta se almacenan los archivos de javascript mediante los cuales se manejan los eventos en la aplicación tales como animaciones y comunicación con servicios web.
  1. **CARGUE DE LA APLICACIÓN EN GITHUB.**

1. Ingrese a su cuenta de GitHub con su usuario y clave.
2. Descargue e instale la aplicación GitHub para Windows desde: <https://help.github.com/articles/set-up-git>.

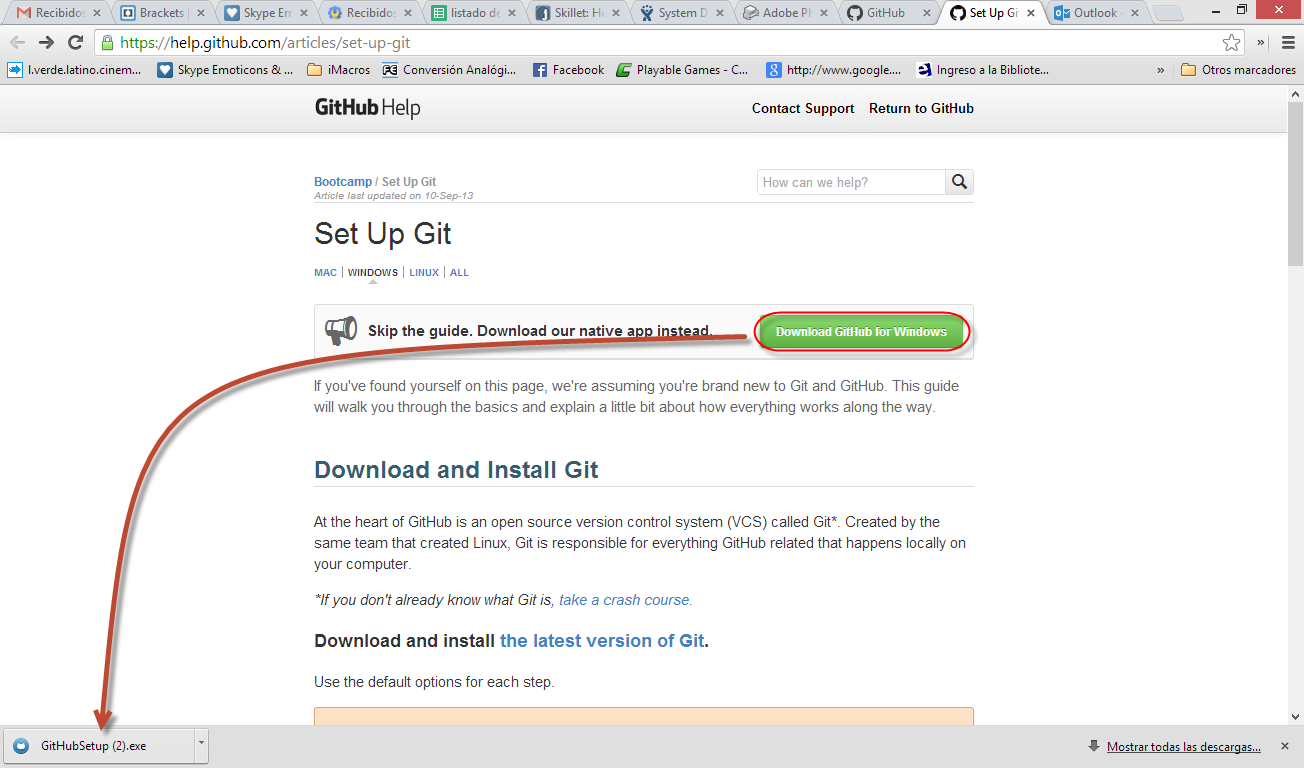


Ilustración 10. Descargar Github for Windows

1. Ejecute la aplicación github y regístrese con su usuario y clave de github. Después de esto, ubíquese en la sección izquierda de la ventana y seleccione la opción “repositories” bajo el título **local**.

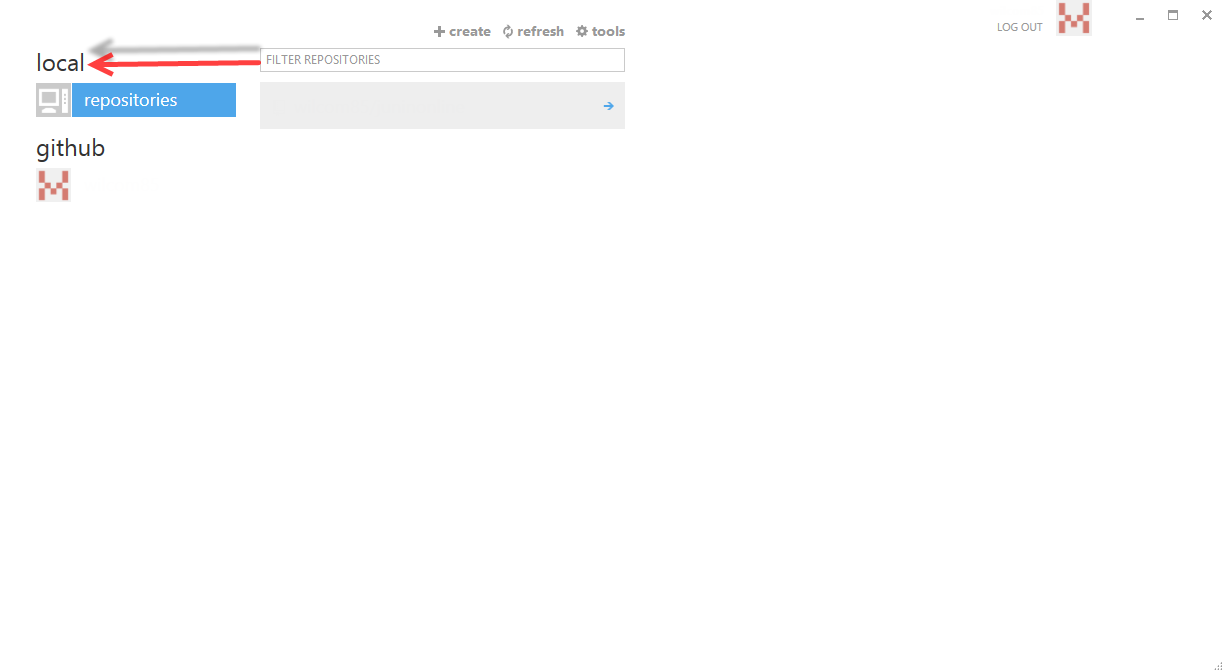


Ilustración 11.Repositorio local GitHub

1. De clic sobre la opción “Create” para crear un nuevo repositorio local. Diligencie el formulario teniendo en cuenta las siguientes indicaciones.
   * 1. Al repositorio se le debe dar el mismo nombre de la carpeta que contiene los archivos de la aplicación.
     2. En el campo No. 2 va la descripción del contenido del repositorio.
     3. En el campo número 3 va la ruta a la carpeta que contiene la aplicación.
     4. Marque el check “putt o gitHub” para que el repositorio se cree en la nube de Github.
     5. La opción “keep this code private” permite que el código que se carga en gitHub permanezca privado. Esta opción solo está disponible para cuentas con plan de pago.
2. Después de haber diligenciado el formulario, de clic en el botón crear.

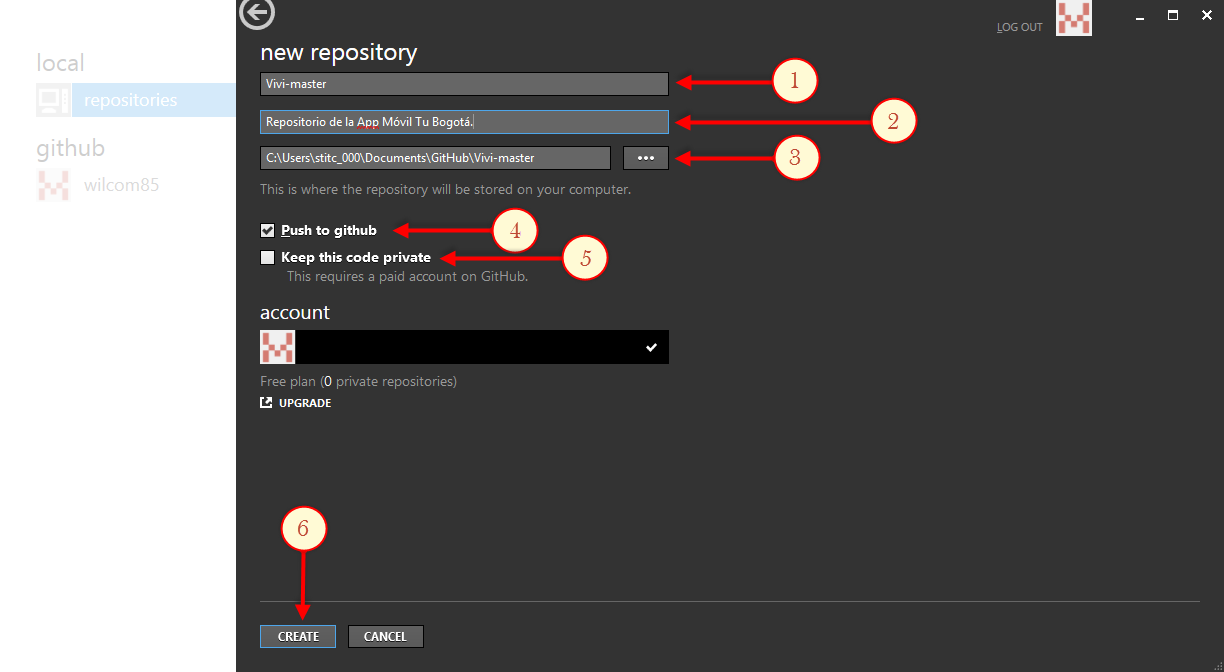


Ilustración 12. Formulario de creación de repositorio.

1. En pantalla aparece el nuevo repositorio creado.

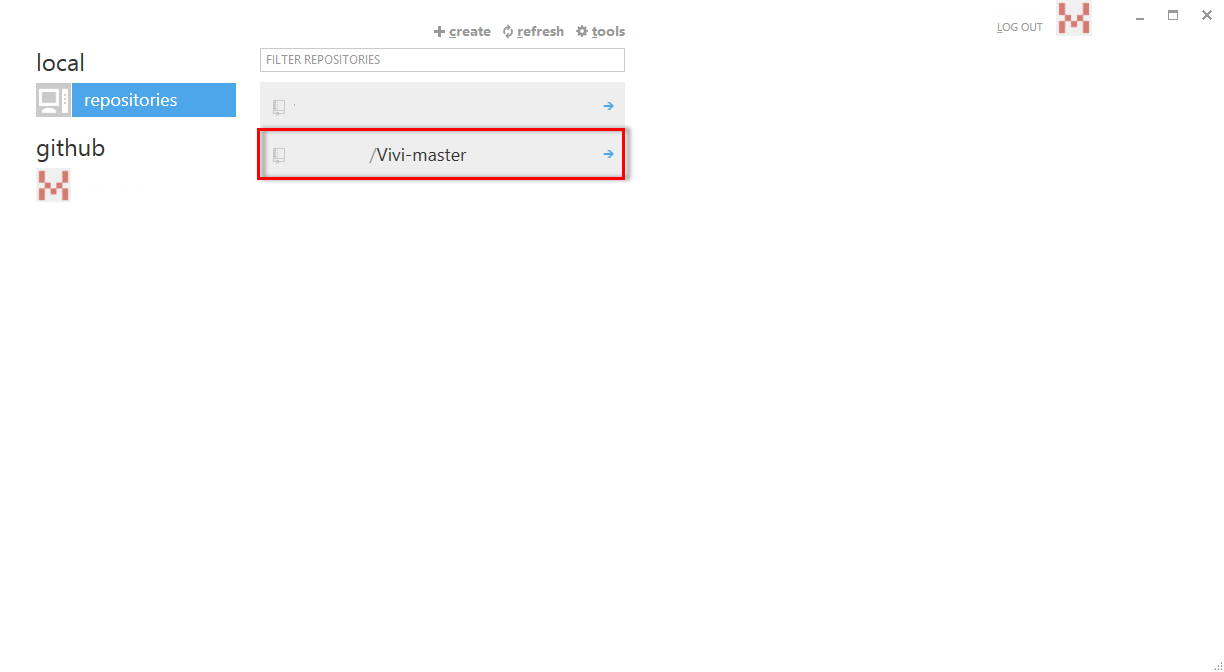


Ilustración 13. Nuevo repositorio creado

1. De clic sobre el ícono  que se encuentra al lado del nombre del nuevo repositorio para conocer su contenido. La carga de esta información puede tomar unos minutos.
2. En pantalla aparece el listado de archivos de la aplicación desarrollada en html5.

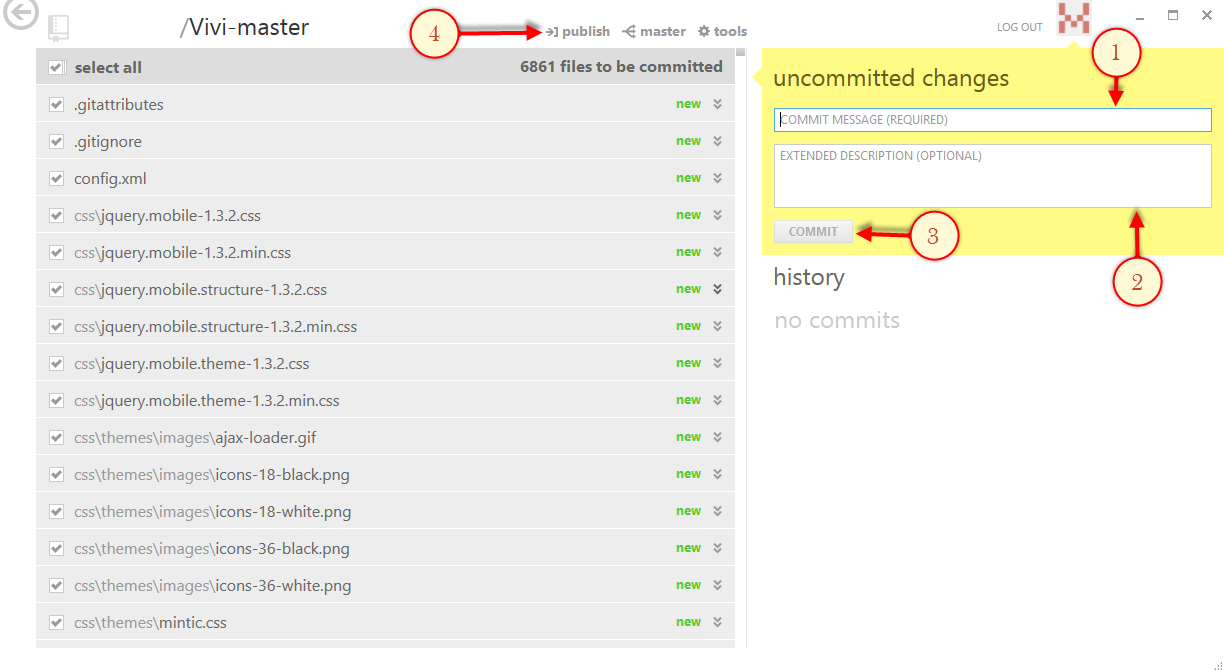


Ilustración 14. Lista de archivos para cargar en el repositorio.

1. Diligencie el formulario que aparece resaltado en color amarillo en la parte superior derecha de su pantalla. En este formulario deberá especificar los cambios que se están cargando en el repositorio indicando un título en el primer campo y una descripción extendida.
2. De clic en el botón “COMMIT” para cargar los archivos que se van a actualizar en el repositorio GitHub en la nube.
3. Finalmente de clic en la opción “publish” para cargar los archivos de la aplicación en el repositorio GitHub.
   1. **REGISTRAR UDID DE DISPOSITIVOS APPLE.**

Para poder realizar pruebas sobre una aplicación de iOS que no ha sido publicada, es necesario tener una suscripción a Apple como iOS developer. Para registrar el UDID de un dispositivo Apple de pruebas, realice el siguiente procedimiento:

1. Regístrese con su usuario y clave de Apple developer en: <https://developer.apple.com>.
2. En la pantalla inicial, seleccione la opción **“Certificates, Identifiers & Profiles”**.

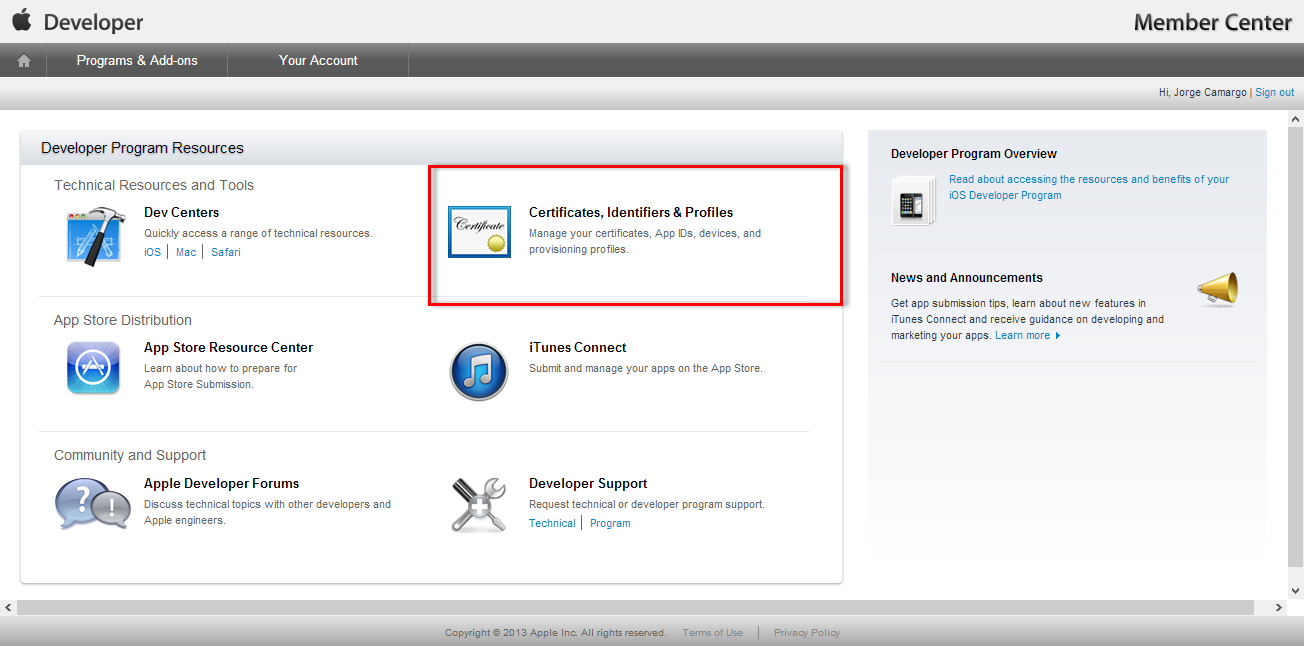


Ilustración 15. Certificados identificadores y perfiles.

1. En la pantalla que le muestra el sistema, identifique el apartado de iOS Apps y en este, seleccione la opción Devices.

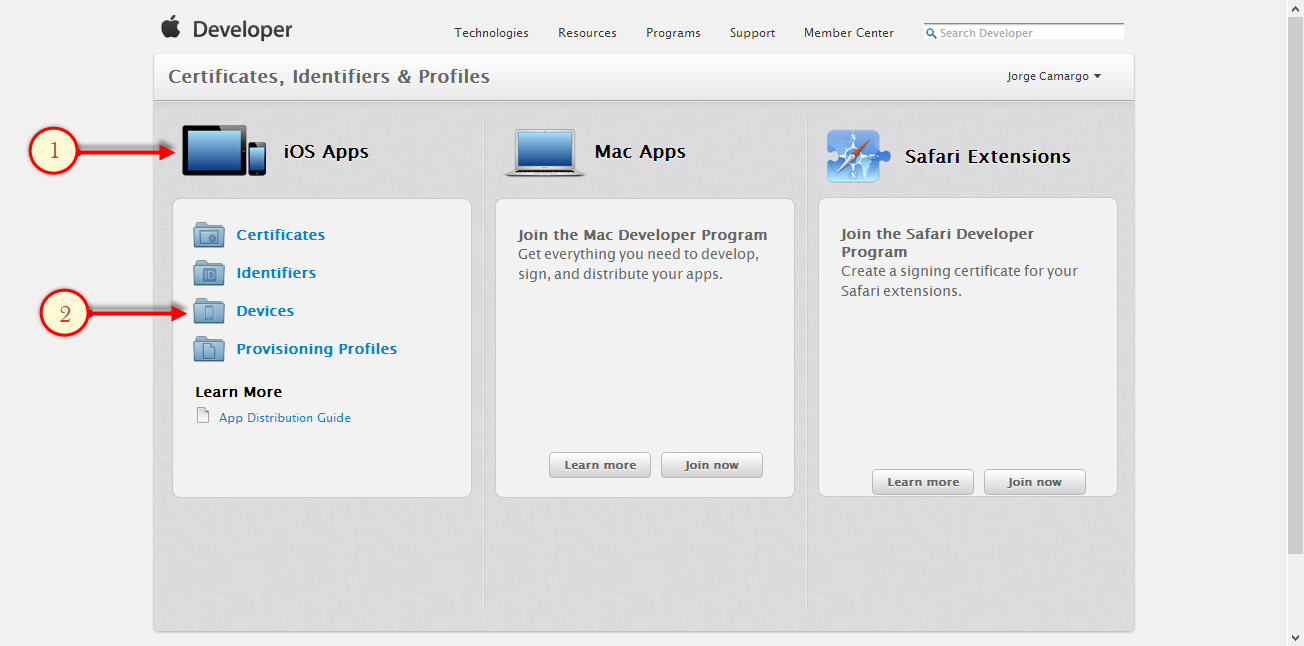


Ilustración 16. IOS Apps

1. Apple le muestra una pantalla en la cual aparece la interfaz para el registro de nuevos dispositivos UDID.
2. Haga clic en el botón + para adicionar un dispositivo.

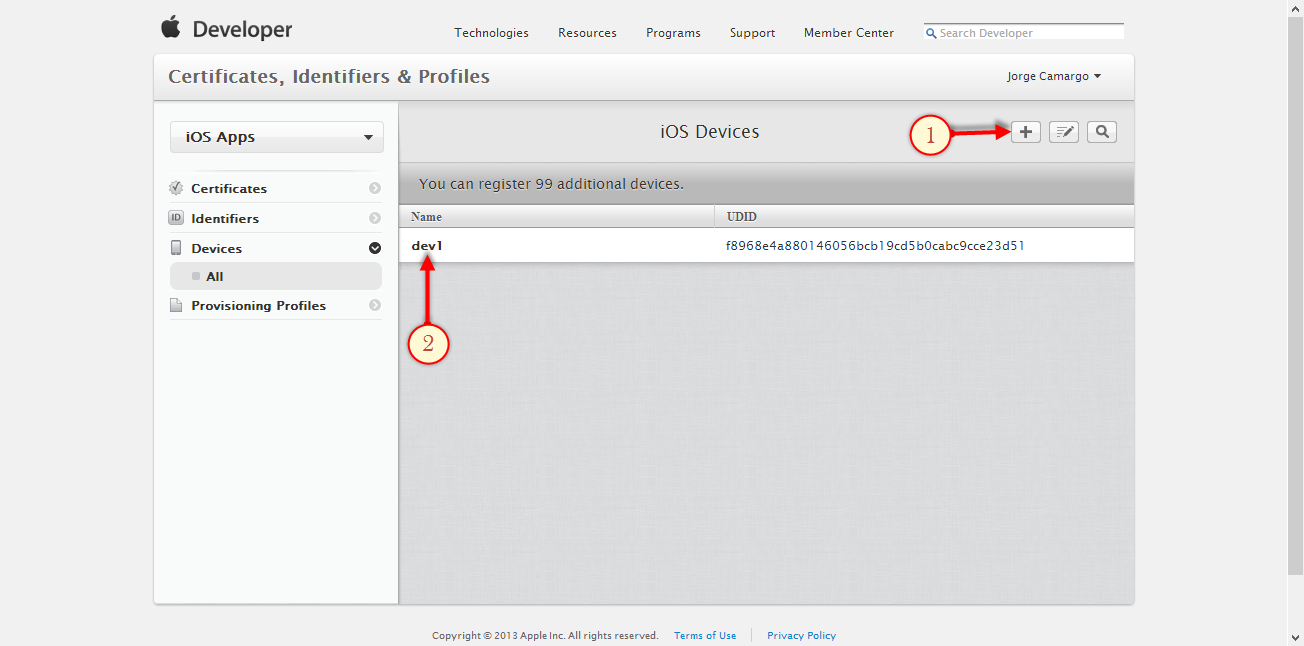


Ilustración 17. Registrar UDID de dispositivos IOS

1. Aparece un formulario en el cual Ud. Debe diligenciar el nombre del dispositivo y el UDID.

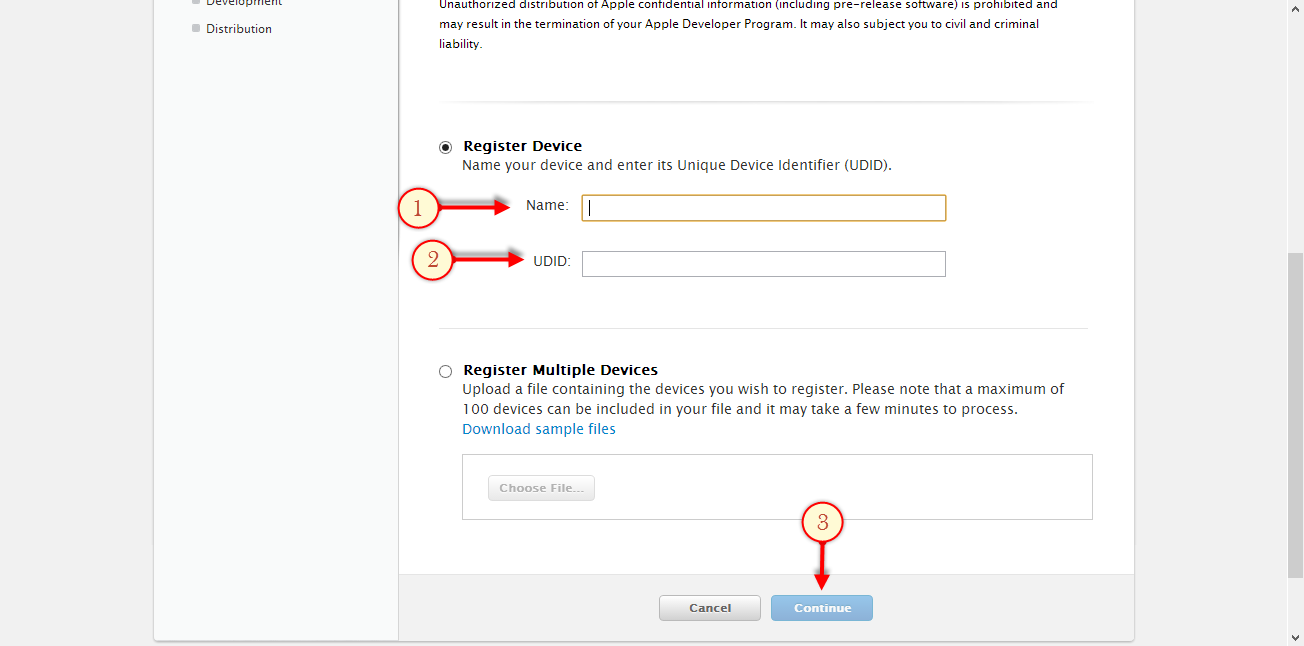


Ilustración 18. Formulario de registro -Nuevo UDID

1. En el campo Name, diligencie el nombre que quiere asignar al dispositivo. En el campo UDID diligencie el código UDID del dispositivo que va a registrar. Luego de clic en el botón “**Continue**”.
2. En pantalla aparece la información del nuevo dispositivo. De clic en el botón “**Register**” para terminar el registro.

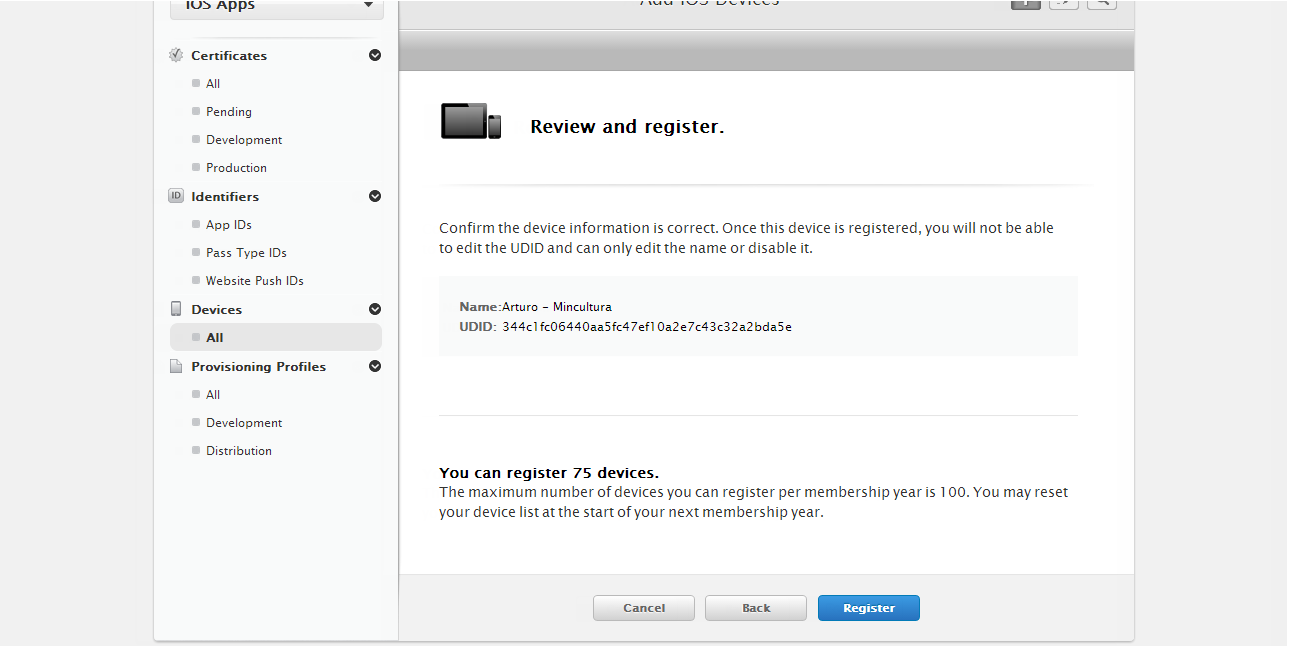


Ilustración 19.Información del dispositivo

1. Aparece en pantalla un mensaje que indica que el registro se completó. De clic en “**Done”**.

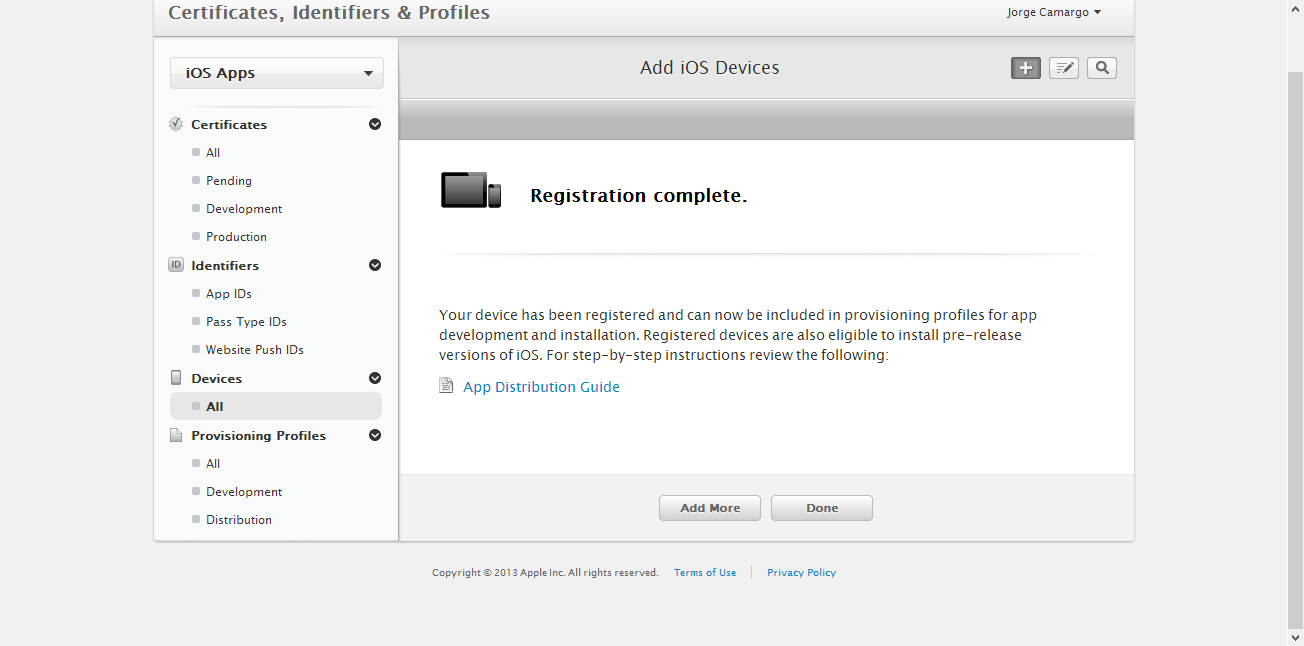


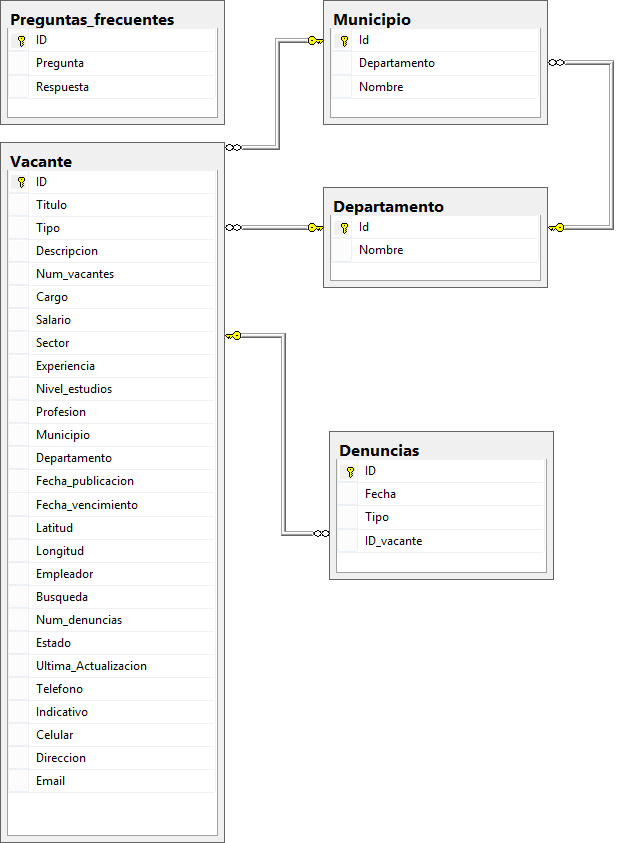
Ilustración 20. Registro completo

El UDID del nuevo dispositivo aparecerá en la lista de dispositivos asociados a su cuenta.

1. MI - PROCESO DE CREACIÓN Y CONFIGURACIÓN DE BASE DE DATOS

El desarrollo móvil y el sitio web utilizado para la Aplicación Servicio de empleo móvil requiere la creación y configuración de bases de datos de la siguiente

Manera.



1. MI - INSTALACIÓN DE LA APLICACIÓN
   1. **AMBIENTE** **DE PRUEBAS**
   2. **INSTALACIÓN DE LA APLICACIÓN ANDROID EN AMBIENTE DE PRUEBAS**
2. El desarrollador compila la aplicación y genera un archivo con extensión .apk.
3. Descargue el archivo en un dispositivo Android.

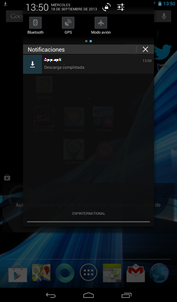


Ilustración 21.Descarga del aplicativo

1. El dispositivo reconoce el archivo como aplicación y muestra las opciones para su instalación.

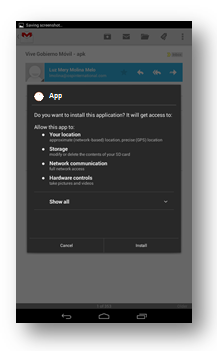


Ilustración 22. Pantalla de instalación del aplicativo.

1. Seleccione la opción Instalar.
2. La aplicación se instala en el dispositivo móvil.

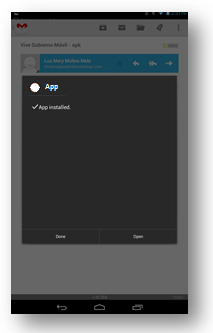


Ilustración 23. Mensaje de aplicación instalada

* 1. **INSTALACIÓN DE LA APLICACIÓN iOS EN AMBIENTE DE PRUEBAS**

1. El desarrollador compila la aplicación y genera un archivo con extensión .ipa.
2. Descargue el archivo en el disco de su computador personal.
3. Ejecute la aplicación iTunes en su computador personal.
4. Seleccione el archivo .ipa de su sistema de carpetas y arrástrelo hacia el panel principal de iTunes.

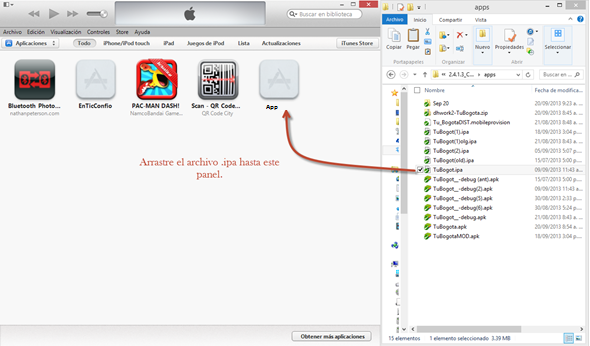


Ilustración 24. Arrastrar archivo .ipa a iTunes

1. Conecte el dispositivo Móvil y espere a que se sincronice con iTunes.
2. Cuando termine la sincronización, de clic en el botón con el nombre de su dispositivo (iPad, iPhone o iPod)



Ilustración 25. Ver información del dispositivo

1. En la ventana que se despliega, seleccione la opción Aplicaciones en el panel de menú.
2. Se despliega la lista de aplicaciones que están instaladas o pendientes pon instalar. Busque la aplicación que se adicionó previamente a itunes en dicha lista.
3. De clic en el botón instalar de la aplicación que se adicionó previamente en iTunes.
4. Finalmente de clic en el botón Sincronizar de iTunes. La aplicación se instala en el dispositivo móvil.

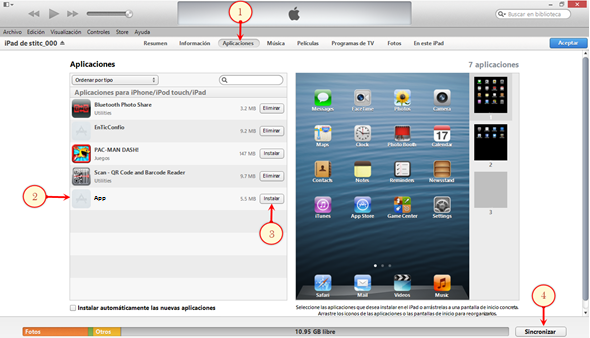


Ilustración 26.. Instalar aplicación desde iTunes

**Nota:** Recuerde que para poder instalar la aplicación correctamente, el UDID del equipo en el que se está instalando la aplicación debe estar registrado en la cuenta del desarrollador.

* 1. **AMBIENTE DE PRODUCCIÓN**

**30.5 INSTALACIÓN DE LA APLICACIÓN ANDROID EN AMBIENTE DE PRODUCCIÓN**

1. Desde su dispositivo Android, seleccione la aplicación **Google Play.**



1. Seleccione el botón de buscar y escriba el nombre de la aplicación.
2. Aparece en pantalla un listado con las aplicaciones que corresponden con el nombre buscado.
3. Seleccione la aplicación. En pantalla aparece la descripción de la aplicación y la opción instalar.
4. De clic en el botón instalar. La aplicación se descarga y se instala en el dispositivo automáticamente.

* 1. **INSTALACIÓN DE LA APLICACIÓN iOS EN AMBIENTE DE PRODUCCIÓN**

1. Ingrese al AppStore de Apple.
2. Busque la aplicación utilizando la opción “Buscar”.
3. Seleccione el botón Free de la aplicación.
4. Seleccione el botón “Install App”.
5. La aplicación se descarga y se instala automáticamente en su dispositivo.
6. MI - VERIFICAR INSTALACIÓN

31.1 VERIFICAR INSTALACIÓN ANDROID

1. Después de instalar la aplicación, en su dispositivo Android seleccione el botón de Aplicaciones.
2. Busque la Aplicación Servicio de empleo y ejecútela.
3. Al iniciar la aplicación, debe aparecer un splash que muestra el logo de la entidad y el nombre de la aplicación mientras esta se carga.
4. Cuando se carga la aplicación, debe aparecer en pantalla el home donde se muestra el menú principal.

**31.2 VERIFICAR INSTALACIÓN iOS**

1. Después de instalar la aplicación, en su dispositivo iOS, busque la Aplicación Servicio de empleo móvil y ejecútela.
2. Al iniciar la aplicación, debe aparecer un splash que muestra el logo de la entidad y el nombre de la aplicación mientras esta se carga.
3. Cuando se carga la aplicación, debe aparecer en pantalla el home donde se muestra el menú principal.
4. MI - PROBLEMAS FRECUENTES
   1. **PROBLEMAS DEL DISPOSITIVO MÓVIL**
5. El dispositivo móvil no cuenta con espacio suficiente para instalar la aplicación.

**Solución:**

Desinstale alguna de las aplicaciones que ya no use antes de instalar la nueva aplicación.

1. MI - INSTALADORES
   1. **INSTALADOR DE LA APLICACIÓN EN TIENDA ANDROID.**

La versión android de la aplicación para el reto Insumos Agropecuarios se identifica como “empleo.apk” la cual se puede descargar desde la tienda de aplicaciones de google, identificada como Play Store. La instalación debe hacerse en dispositivos móviles con sistema operativo Android.

El usuario debe tener una cuenta en la tienda de aplicaciones para que pueda realizar el descargue de la aplicación.

* 1. **INSTALADOR DE LA APLICACIÓN EN TIENDA IOS.**

La versión iOS de la aplicación para el reto Isumos Agropecuarios se identifica como “empleo.ipa” la cual se puede descargar desde la tienda de aplicaciones App Store. La instalación debe hacerse en dispositivos móviles con sistema operativo iOS.

El usuario debe tener una cuenta en la tienda de aplicaciones para que pueda realizar el descargue de la aplicación.

MANUAL DE USO Y OPERACIÓN

34. MU - OBJETIVOS DEL SISTEMA

Permitir a los ciudadanos comparar, analizar y reportar los precios de los insumos agropecuarios de Colombia, dándoles la posibilidad de obtener un precio justo y ayudar a ejercer control en los puntos de venta que no cumplen las regulaciones de precios, a través de herramientas móviles usando las plataformas de mayor penetración en el mercado como lo son IOS y Android

35. MU - REQUISITOS DEL SISTEMA

Dispositivo móvil con sistema operativo Android versión 2.3 o superior y con conexión a internet.

Dispositivo móvil con sistema operativo iOS (iPad/iPhone), con versión iOS 6.X o superior y con conexión a internet.

36. MU - DESCRIPCIÓN DEL SISTEMA

**36.****1 ADMINISTRADOR**

El administrador estará encargado de realizar el mantenimiento de los datos que provee la aplicación móvil, en la base de datos que alimenta el servicio web que consume la aplicación.

**36.2 USUARIO FINAL**

El usuario tendrá en la Aplicación Servicio de Empleo móvil, una herramienta que le permitirá consultar las oportunidades de empleo, en caso de empleador podrá registrar y modificar diferentes ofertas de empleo.

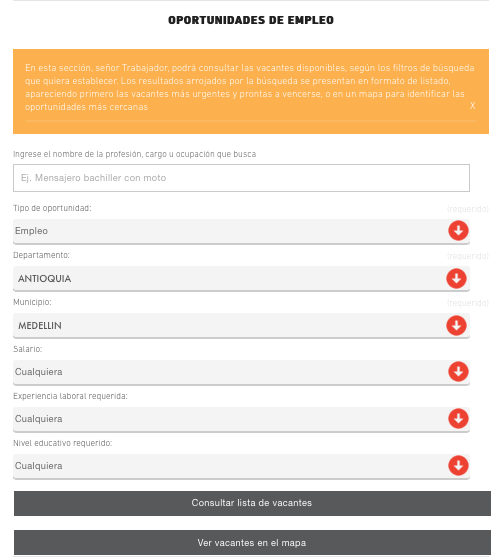
37. MU - MANUAL DE USO

**37.1 MENÚ PRINCIPAL**

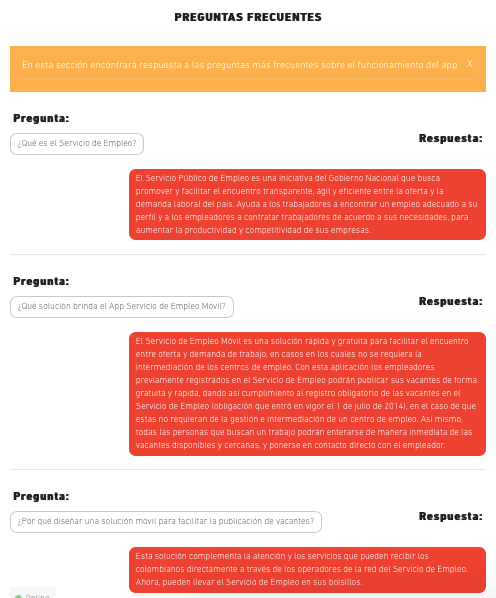
La Aplicación cuenta con un menú de navegación para acceder a todas las opciones de la aplicación



**37.3 Oportunidades de empleo**



**37.4 preguntas frecuentes**

****

1. MU - MANUAL DE OPERACIÓN

E

l modelo de operación es un marco de trabajo que define todas las actividades de intervención humana y los roles asociados que están involucrados en la continuidad y operación de un sistema de información.

La presente sección es una herramienta de consulta en la cual se describen las principales operaciones requeridas para que la Aplicación Servicio de empleo móvil funcione de acuerdo a los requerimientos no funcionales y entregue de manera correcta su funcionalidad a los usuarios finales.

* 1. **AMBIENTES DE DESARROLLO**

Para desarrollar la aplicación se empleó el framework PhoneGap, el cual permite la creación de aplicaciones móviles en distintas plataformas (iOS y Android para este caso) empleando solo una base de código fuente. La herramienta PhoneGap corresponde a un software libre.

Para el desarrollo, el lenguaje de programación empleado fue JavaScript empleando la librería JQuery .

* 1. **LICENCIAS PARA PUBLICACIÓN DE LAS APLICACIONES**

La siguiente tabla relaciona los requisitos de permisos en las tiendas para la operación de la aplicación y el vencimiento.

Tabla11. Licencias requeridas

| ELEMENTOS | VENCIMIENTO | ESQUEMA DE MANTENIMIENTO | OBSERVACIONES |
| --- | --- | --- | --- |
| Cuenta Google Play Store. | Ninguno | Ninguno | Pago único de 25 dólares |
| Cuenta Apple para iOS. | 1 año | No Aplica | Pago anual de 99 dólares |
| Servidor web | Ninguno | Ninguno |  |

* 1. **PLAN DE CONTINGENCIA**

El plan de contingencia para la Aplicación Servicio de empleo móvil consiste en a la verificación de las configuraciones que permitan garantizar la correcta publicación de las aplicaciones en los canales de distribución definidos, que son en este caso las tiendas de aplicaciones de la respectiva plataforma, así como que el dispositivo móvil del ciudadano cumpla con las condiciones mínimas necesarias para la descarga e instalación de las soluciones.

En caso de que los ciudadanos no encuentren la aplicación publicada en las tiendas, la entidad deberá verificar los siguientes aspectos:

* Corroborar el estado de suscripción a las cuenta de desarrollador tanto de Apple como de Google. En caso de que las cuentas no hayan sido renovadas en el término anual, las aplicaciones serán suspendidas de las respectivas tiendas.
* Verificar la configuración de firmas y correcto despliegue de las aplicaciones, así como el estado de los servicios web. Los fabricantes se reservan el derecho de suspender aplicaciones que los usuarios hayan reportado con errores, a razón de problemas tales como el estado de sus servicios, o versiones que presenten fallos graves y que pongan en riesgo el desempeño del dispositivo.

Adicionalmente, si el ciudadano encuentra la aplicación y no la puede descargar o actualizar, deberá tener en cuenta lo siguiente:

* Verificar que su dispositivo móvil cumpla con los requerimientos mínimos de instalación de las aplicaciones móviles desarrolladas.
* Verificar que no esté accediendo desde un dispositivo móvil que ha sido alterado o programado para obtener privilegios de administrador, o pirateado para la descarga ilegal de aplicaciones. En estos casos no se puede garantizar la correcta instalación o funcionamiento de las soluciones.
* Verificar que la cuenta de usuario asociada al dispositivo corresponda a una cuenta colombiana, dado que la aplicación puede no ser desplegada a nivel internacional, sino únicamente a nivel nacional.
* Si el usuario no puede actualizar su aplicación a una nueva versión, puede deberse a que su dispositivo o sistema operativo haya quedado obsoleto para la nueva versión publicada por la entidad. De otro modo el ciudadano deberá revisar lo mencionado en los incisos anteriores.

* 1. **ACTIVIDADES DE MONITOREO DE LOS SERVICIOS**

En esta sección se presentan todas las actividades y procedimientos que se deben ejecutar sobre herramientas para evidenciar que el sistema funciona correctamente.

Debido a las características de la aplicación se requiere realizar el monitoreo en las tiendas Google Play y App Store.

Igualmente debido a la necesidad de consumo del web service de Open Data es necesario que la entidad, realice las actividades requeridas para asegurar que la información se encuentre actualizada y disponible.

1. TERMINOLOGÍA

**Android**: Es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google. Este sistema por lo general maneja aplicaciones como Google Play.

**HTML5**: Quinta versión del lenguaje HTML (HyperText Markup Language o lenguaje de marcado hipertextual), el cual se utiliza predominantemente para el desarrollo de páginas web.

**iOS:** (Anteriormente denominado iPhone OS): Es un sistema operativo móvil de Apple. Originalmente desarrollado para el iPhone, siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV. Apple, Inc. no permite la instalación de iOS en hardware de terceros.

**Java**: Lenguaje de programación de propósito general, concurrente, basado en clases, y orientado a objetos

**JSON**: JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

**Objective-C**: Lenguaje de programación orientado a objetos creado como un superconjunto de C. Se usa como lenguaje de programación en Mac OS X, iOS y GNUstep

**Servicios Web**: Un servicio web (en inglés, Web services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles.

**XML**: extensible Markup Language (lenguaje de marcas extensible), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. Permite definir la gramática de lenguajes específicos para estructurar documentos grandes; da soporte a bases de datos, por lo cual es útil cuando varias aplicaciones se deben comunicar entre sí o integrar información.

1. ANEXOS

**ANEXO 1 - PAUTAS PARA LA CODIFICACIÓN DEL CÓDIGO FUENTE**

El uso de un estándar para el código es fundamental para el desarrollo de software, de tal manera que se garantice la mantenibilidad del mismo en el tiempo. Una buena guía permite a los desarrolladores escribir programas usando un conjunto de reglas del lenguaje, estándar de nombrado de variables, manejo de excepciones, entre otros. A continuación se describen las guías de codificación para el desarrollo de aplicaciones móviles en Android e iOS.

**Guía de codificación para Android**

La siguientes reglas de codificación están basadas en el estándar que los programadores del sistema operativo Android deben seguir (Code Style Guidelines for Contributors#). Esta guía es utilizada por miles de programadores en el mundo que hacen parte de la comunidad que desarrolla dicho sistema operativo.

**Reglas del lenguaje Java**

Evite escribir código que ignora el manejo de excepciones como sucede a continuación:

void setServerPort(String value) {

try {

serverPort = Integer.parseInt(value);

} catch (NumberFormatException e) { }

}

Por el contrario, maneje las excepciones de manera explícita como se describe a continuación:

void setServerPort(String value) throws NumberFormatException {

serverPort = Integer.parseInt(value);

}

Evite capturar excepciones genéricas como en el siguiente ejemplo:

try {

someComplicatedIOFunction(); // may throw IOException

someComplicatedParsingFunction(); // may throw ParsingException

someComplicatedSecurityFunction(); // may throw SecurityException

// phew, made it all the way

} catch (Exception e) { // I'll just catch all exceptions

handleError(); // with one generic handler!

}

Evite el uso de la instrucción Finalize:

Esta instrucción es una manera de expresar código que se ejecute cuando el recolector de basura elimina el objeto.

**Reglas para importar librerías**

Importe librerías con su nombre calificado de manera explícita: Import foo.Bar;

Evite importar paquetes completos: Import foo.\*;

**Reglas de estilo en Java**

Utilice el estándar de comentarios de Java:

/\*

\* Copyright (C) 2010 The Android Open Source Project

\*

\* Licensed under the Apache License, Version 2.0 (the "License");

\* you may not use this file except in compliance with the License.

\* You may obtain a copy of the License at

\*

\* http://www.apache.org/licenses/LICENSE-2.0

\*

\* Unless required by applicable law or agreed to in writing, software

\* distributed under the License is distributed on an "AS IS" BASIS,

\* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

\* See the License for the specific language governing permissions and

\* limitations under the License.

\*/

package com.android.internal.foo;

import android.os.Blah;

import android.view.Yada;

import java.sql.ResultSet;

import java.sql.SQLException;

/\*\*

\* Does X and Y and provides an abstraction for Z.

\*/

public class Foo {

...

}

Cada clase o método no trivial debe tener sus respectivos comentarios en Javadoc describiendo su lo que el método o clase hace:

/\*\* Returns the correctly rounded positive square root of a double value. \*/

static double sqrt(double a) {

...

}

**Nombres cortos para los métodos**

Escriba nombres de métodos cortos. En algunos casos es útil que el nombre de un método sea largo, sin embargo, trate de usar nombres que no excedan los 40 caracteres.

**Definición de variables**

Defina las variables al comienzo al comienzo de una clase o método, o inmediatamente antes de su llamado en los métodos.

**Alcance de las variables**

Limite el alcance de las variables. Las variables locales deberían ser declaradas en el punto en donde van a ser utilizadas. Cada variable debería tener una inicialización.

**Importación de paquetes**

El orden para importar paquetes debería seguir el siguiente:

• Paquetes de Android

• Paquetes de terceros

• Java y javax

**Indentación**

Utilice espacios para indentar el código: 4 espacios para indentar bloques y 8 líneas para indentar llamados a funciones y asignaciones.

Instrument i =

someLongExpression(that, wouldNotFit, on, one, line);

**Convención para nombres de variables**

Siga convenciones para el nombre de variables

• Nombres de variables no públicos y no estáticos deben empezar con la letra m.

• Variables estáticas deben empezar con s.

• Variables tipo public, static y final, deben ser nombradas en mayúsculas y unidas con underscore.

public class MyClass {

public static final int SOME\_CONSTANT = 42;

public int publicField;

private static MyClass sSingleton;

int mPackagePrivate;

private int mPrivate;

protected int mProtected;

}

**Manejo de corchetes**

Utilice un estilo estándar para el manejo de corchetes:

class MyClass {

int func() {

if (something) {

// ...

} else if (somethingElse) {

// ...

} else {

// ...

}

}

}

**Longitud de una línea**

Cada línea de código debería ser de máximo 100 caracteres de longitud. En caso que la línea sea de un comentario con comando de ejemplo o una URL de más de 100 caracteres, utilice los caracteres que necesite por claridad.

**Uso de anotaciones**

Las anotaciones deberían preceder otros modificadores para el mismo elemento del lenguaje. Anotaciones simples como @Override pueden ser listadas en la misma línea con el elemento del lenguaje. Si hay múltiples anotaciones o anotaciones parametrizadas, ellas deberían cada una ser listada una por línea en orden alfabético.

Las prácticas estándar de Android para las tres anotaciones predefinidas en Java son:

• @Depecated

• @Override

• @SuppressWarnings

**Tratamiento de acrónimos como palabras**

Trate los acrónimos y abreviaciones como palabras en nombres de variables, métodos y clases. Los nombres son mucho más sencillos de leer:

Bien Mal

XmlHttpRequest XMLHTTPRequest

getCustomerId getCustomerID

class Html class HTML

String url String URL

long id long ID

**Uso de comentarios TODO**

Use comentarios TODO para el código que es temporal, soluciones temporales, o código bueno pero no lo suficiente. Los TODOs deberían incluir la cadena TODO en mayúsculas, seguido de dos puntos:

// TODO: Remove this code after the UrlTable2 has been checked in.

**Manejo de Logs**

El manejo de logs es necesario pero tiene un impacto fuerte en el desempeño de una aplicación y rápidamente pierden su utilidad durante el desarrollo de una aplicación. La generación de logs ofrece cinco niveles diferentes:

• ERROR

• WARNING

• INFORMATIVE

• DEBUG

• VERBOSE

Utilice el tipo de log más adecuado según las necesidades.

**Guía de codificación para iOS**

Apple ofrece a la comunidad de desarrolladores en Objective-C# y Cocoa#. A continuación se enumeran las reglas que deben tenerse en cuenta en la codificación de aplicaciones móviles para iOS.

**Manejo de nombres**

Cuando se declare una variable, su nombre debería ser lo suficientemente descriptivo, como por ejemplo:

NSInteger currentColumn;

**Nombres de archivos**

Los nombres de archivos deberían ser descriptivos, iniciando con una letra en mayúscula. Cuando se crean las clases en Xcode, este es el comportamiento por defecto, dado que los nombres de las clases corresponden con los nombres de archivos:

JPClassName.h

JPClassName.m

**Nombres de clases**

Los nombres de clases deben ser indicativos de su jerarquía de clases. E decir, a partir del nombre de una clase usted debería ser capaz de correctamente inferir el tipo de clase que ésta es.

JPClassName.h

JPClassName.m

El anterior nombre es descriptivo e inmediatamente le indica al desarrollador que esta clase es de tipo TableViewController, la cual muestra productos.

**Protocolos**

Los protocolos deberían ser nombrados de la misma manera en que se nombran las clases a las que pertenecen, adicionalmente añadiendo el rol del protocolo. Si no hay un rol distinto, añadir Protocol es suficiente. El objetivo es ser capaz de decir en una simple mirada lo que el símbolo representa.

JPClassName.h

JPClassName.m

**Categorías**

Los nombres de las categorías deberían indicar el nombre de la clase que está siendo extendida, seguido por un +, seguido por el nombre de la categoría.

NSString+JP1337Additions.h/m

Las categorías son utilizadas para extender clases en tiempo de ejecución sin necesidad de usar subclassing. Sin embargo las categorías no deben ser usadas para sobrescribir métodos existentes.

**Variables**

Mantenga las declaraciones de variables en su propia línea aún si ellas son del mismo tipo.

NSString \*localString;

NSString \*password;

Las variables de instancia deberían ser nombradas como las variables normales. Las variables de instancia no deberían ser declaradas en la interfaz pública de la clase (@interface). En su lugar, declárelas en un bloque privado (@interface JPClassName()).

**Constantes**

Las constantes deberían ser declaradas con la palabra reservada const y deberían se nombradas iniciando con la letra k, seguido del más cercano tipo o nombre de clase, seguido del valor:

NSString \*const kJPProductsDataKey; //defined in an implement. file

**Métodos**

Los nombres de métodos deberían ser descriptivos, en donde la claridad es la característica más importante:

- (id)initWithFrame:(CGRect)frame; // OK

- (id)initWithFrame:(CGRect)frame backgroundColor:(UIColor \*)color; // OK

- (id)initWithFrame:(CGRect)frame andBackgroundColor:(UIColor \*)color; // bad