



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

Universidad Nacional Autónoma de México

Facultad de Artes y Diseño

La presencia del Diseño Gráfico en la creación de apps.

Tesis

Que para obtener el Título de:

Licenciado en Diseño y Comunicación Visual

Presenta: Juan Martín León Piña

Director de Tesis: Maestro Juan Carlos Miranda Romero

México, D.F., 2014



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A mi **papá y mi mamá (mi ángel)** por la educación que me han dado, por siempre exigirme dar lo mejor de mi, por enseñarme a salir adelante a pesar de las adversidades y por ser siempre mi ejemplo a seguir.

A mi **hermana Karla** por estar ahí en todo momento, por ser mi amiga y consejera, por cuidarme y ser paciente conmigo.

Al resto de **mi familia** por estar siempre unida y apoyarme en todo momento.

A **mis amigos** por sus buenas vibras y sus palabras de aliento.

ÍNDICE

INTRODUCCIÓN

13

CAPÍTULO 1

APLICACIONES MÓVILES

Antecedentes históricos
¿Qué es una aplicación móvil?
Tipos de apps
Plataformas de desarrollo

27

CAPÍTULO 2

METODOLOGÍA DEL DISEÑO

Metodología
Metodologías del Diseño
 Christopher Jones
 Bruno Munari
 Bernd Löbach
 Gui Bonsiepe
 Morris Asimow

39

CAPÍTULO 3

GRÁFICOS PARA APPS

Color
Tipografía
Resoluciones de Pantalla
Íconos
Formatos de Imagen

55

CAPÍTULO 4

DESARROLLO DE APPS

Idea
Análisis
Arquitectura de la información
Diseño Gráfico
Desarrollo
Prototipo
Evaluación
Lanzamiento en tiendas
Resultado final

95

CONCLUSIONES

97

FUENTES DE CONSULTA

INTRODUCCIÓN

En los últimos años, la industria de los dispositivos móviles ha tenido un gran impacto en la tecnología, año tras año vemos nuevos lanzamientos, sean smartphones o tablets, los cuales presentan numerosas novedades en hardware; además semana a semana surgen nuevas aplicaciones para estos aparatos que pretenden resolver distintas necesidades del hombre. Actualmente, el número de teléfonos inteligentes vendidos ronda los 2 mil millones y sigue en aumento; es, sin lugar a dudas, el sector que mayor innovación y expectativa está generando y generará. Aunque parecen un fenómeno reciente, la realidad es que los dispositivos móviles y las aplicaciones hace algunos años que están entre nosotros y como todo, necesitan de tiempo y distintos factores para llegar a ser lo que son ahora.

Los factores que han logrado el auge de esta industria son: mejoras en las características de hardware de los dispositivos móviles con la creación de pantallas táctiles capacitivas y resistivas; el uso generalizado de apps para diversas actividades del día a día, cubriendo con ello las necesidades de más usuarios. La diversidad de Sistemas Operativos y dispositivos llegando a nuevos aparatos tales como televisores, lentes, relojes y automóviles, sin dejar a un lado teléfonos y tablets; por último la accesibilidad con que se producen y distribuyen nuevas apps ha motivado a los desarrolladores a seguir creando este tipo de software, pues los alcances de sus productos son cada vez mayores a través de las tiendas de aplicaciones presentes en distintos países del mundo.

Siendo concientes de nuestra realidad, los dispositivos móviles se han convertido en una herramienta de la vida cotidiana, que ayudan con las tareas diarias del hogar, trabajo o el entretenimiento gracias a las apps, permitiendo mayor comodidad y portabilidad al utilizar únicamente un solo objeto para que se puedan realizar distintas actividades; los tiempos en que era necesario cargar con la cámara fotográfica, el reproductor de audio más el celular por citar un ejemplo han quedado en el pasado.

Tal ha sido el éxito y su uso que existe una sobreoferta de apps en las tiendas de aplicaciones móviles; cualquier ingeniero en computación quiere lanzar su aplicación e incluso la gente busca tutoriales para aprender a desarrollar. Si bien muchas de ellas son novedosas carecen de buena funcionalidad y que decir de la interfaz, gráficos poco agradables y sin razón de ser, además de poca usabilidad.

Entendiéndose que no solo se trata de programar, también es importante pensar en el usuario y la interacción que tendrá con la aplicación, que tipo de experiencia de usuario (UI) es la que se le quiere brindar, ya que finalmente es el quien juzga si es buena o mala la aplicación. Al desarrollar, la mayoría de las veces pocos se pone a pensar en como quieren que

la gente vea su aplicación, como será la navegación y la forma en que deben organizarla para tener un buen flujo de información y mejor usabilidad. Por ello es importante tomar en cuenta la labor y ayuda que pueden brindar los diseñadores gráficos en el desarrollo del contenido de cualquier aplicación, quienes buscamos mejores experiencias, apoyadas en el diseño visual que ahora tiene mayor importancia y responsabilidad.

Asimismo, el diseñador debe ser el responsable de crear la identidad de nuestra aplicación, la imagen que queremos darle a la gente; en las tiendas de aplicaciones hay cientos de apps tan parecidos por su función que la única forma de diferenciarlas es por su arte, es decir por sus gráficos y el estilo que estos tienen; si logran impactarnos serán clave en la decisión de que app descargar.

Por supuesto, no se trata de menospreciar el trabajo de los programadores, pues ellos han sido los responsables de desarrollar las apps desde sus inicios, el mercado exige aplicaciones de calidad que puedan destacarse de las demás y no basta solo con que se vean bien, deben cumplir con un buen funcionamiento y ser fáciles de comprender por el usuario.

A través de los siguientes capítulos se analizará el proceso de desarrollo de una aplicación móvil para descubrir el papel importante que juega el diseñador gráfico, para ello primero hay que conocer lo que es una app, los diferentes tipos que existen y cuales fueron sus orígenes. Así como las distintas metodologías del diseño que nos pueden servir para el diseño de interfaz de la aplicación móvil.

Posteriormente se tratan los requerimientos gráficos que debe tener una aplicación, aquellos elementos del diseño que deben ser tomados en cuenta para la interfaz y el arte de ésta. Se hacen especificaciones sobre los colores, tipografía, íconos etc., enfocados a las pantallas de los dispositivos móviles.

Por último hago un estudio sobre el desarrollo de una aplicación, desde la concepción de la idea hasta la publicación en las diferentes tiendas, pasando por el diseño de interacción y de arte. Será aquí donde mayor atención se le preste al trabajo desempeñado por el diseñador gráfico; cabe mencionar que para esta investigación se diseñó la app “Pluzer” como medio para ejemplificar cada paso de este proceso.

Aunque se habla de aplicaciones de manera global, en varias ocasiones se establecen las diferencias entre los sistemas operativos para móviles más comunes: Android y iOS. Se han elegido estas dos plataformas porque hoy en día son las más importantes, por ser las más usadas y quienes mayor compatibilidad tienen con los teléfonos inteligentes.

Además de resaltar la labor del diseñador, esta investigación pretende servir como guía introductoria para quien busque involucrarse en el diseño de apps, debido a que es un tema de reciente aparición, por lo que no existe tanta bibliografía sobre el tema y es muy poco el material en español.

APLICACIONES

MÓVILES

CAPÍTULO I

1.1 ANTECEDENTES

La historia de las aplicaciones móviles no es muy extensa se remite a los últimos años del siglo XX, son el producto de sólo unos pocos años de evolución de la tecnología móvil, desde los primeros teléfonos transportables a los pequeños dispositivos de alta tecnología que tenemos en nuestro bolsillo actualmente. La escala de tiempo tecnológico es completamente diferente a la humana, en dos años de vida humana no hay una evolución significativa, si bien en los primeros años de vida humana es notable, en la edad adulta es difícil de percibir; mientras que en el tiempo tecnológico un año hace ver viejas las cosas, apenas nos estamos familiarizando con algún smartphone cuando ya se hablan de nuevos modelos.

Los primeros esbozos de apps surgen aproximadamente a principios de los años 90, con sencillos softwares contenidos en los celulares de ese entonces, con la función de archivar tus contactos como lo hacen las agendas o también el caso de los ringtones que te permitían seleccionar el sonido de las llamadas entrantes a tu celular. Tal vez una novedad para la época pero eran muy elementales y su diseño era demasiado simple, recordemos que en ese entonces solo había pantallas blanco y negro. Conforme fueron evolucionando los celulares se fueron agregando nuevos programas por ejemplo los juegos de Snake o Tetrix; es hasta que en los celulares comienza a existir la conectividad a Internet con servicios WAP o transferencia de datos que se logra una mayor mejora en estos dispositivos.

Ya en el nuevo siglo diferentes factores ayudaron a que se diera un avance mayor en la industria móvil, como lo menciona Felker D. (2011), diseñador y desarrollador de software, autor de varios libros sobre Android, estos factores fueron: “mayor conocimiento de Internet con redes establecidas por todo el mundo, nuevos movimientos sociales, tecnología de gran alcance involucrada a procesos industriales y mayor apego de la tecnología en el ámbito profesional” (p.10); dando inicio a una nueva etapa de innovación que se vería reflejada en el año 2007 a través de la empresa Apple.



JUEGO DE SNAKE

HISTORIA iOS

El 9 de enero de 2007 dentro del evento Mac World Expo, Steve Jobs fundador de Apple tendría en expectativa a miles de personas que esperaban el nuevo lanzamiento de la com-

pañía y más aun porque los rumores apuntaban hacia la presentación de un teléfono, un sector que la empresa se pensaba desconocía ya que siempre se caracterizo por ser líder en la industria de las computadoras. Los rumores serían confirmados, Steve Jobs hace de manera oficial la presentación del iPhone, producto con el que Apple reinventaría el teléfono según palabras del fundador. Este teléfono fue resultado de años de investigación y desarrollo en los que la empresa fue capaz de combinar perfectamente hardware y software; con ello llegaron las primeras aplicaciones móviles: Mapas, Contactos, Accesorios, Safari (navegador web), etc.



PRIMERA GENERACIÓN DEL IPHONE

El lanzamiento en tiendas del iPhone se dio hasta finales de junio del mismo año, los desarrolladores estaban ansiosos pues querían programar para este teléfono, crear sus propias aplicaciones; pero sus aspiraciones fueron frustradas por Apple debido a que las apps nativas instaladas en el teléfono habían sido programadas exclusivamente por la compañía. En primera instancia

Apple afirmó que no era necesario ya que las aplicaciones para el dispositivo debían ser construidas como aplicaciones web utilizando JavaScript, CSS y HTML. Esto no cayó bien en la comunidad de desarrolladores, querían acceso directo al hardware y la integración con aplicaciones de Apple.

Los desarrolladores buscaron la forma de intervenir los teléfonos con tal de tener aplicaciones nativas, fue así como se hicieron los primeros hackeos al dispositivo, algo que muy pocos creían posible. Alasdair A. (2013), programador de software a medida y constructor de hardware libre, hace referencia a este hecho: “a pesar de que Apple bloqueó el dispositivo debajo, los desarrolladores han tenido acceso a la ingeniería inversa del SDK (Kit de Desarrollo de Software) y se han ido a construir una cadena de herramientas de código abierto que les permite crear aplicaciones nativas para el dispositivo. El llamado “jailbreak”(fuga de la cárcel) lo que permite que se ejecuten aplicaciones de terceros no autorizadas” (p.1).

Esto motivo a que Apple cambiara de opinión y le diera oportunidad a los desarrolladores; en la primavera de 2008 liberaría el kit de desarrollo en una versión beta (SDK), no se sabe si era una estrategia ya prevista por la empresa o realmente se vio obligada a hacerlo.

Por cualquiera de las razones que lo haya hecho resultado ser un éxito, ya que para julio de 2008 se abrió la tienda más grande de aplicación móviles, la Apps Store. Este sitio en línea permitiría descargar millones de aplicaciones gratuitas o con algún costo; tal es el éxito de esta tienda que comienza a hablarse de ella en todos lados, los expertos en esta materia se ven sorprendidos por la gran cantidad de aplicaciones que surgen, así como los millones de descargas de cada una de ellas; Ankeny, J. (2011) desarrollador de software, hace una investigación sobre estos números en el artículo Diez millones de descargas y contando, donde podemos leer que “ App store abre el 11 de julio de 2008 con aproximadamente 550 aplicaciones, incluyendo programas de educación, comercio móvil y las herramientas de productividad de negocios, los juegos representan alrededor de un tercio de la primera ola de aplicaciones, y continuarán dominando todas la categorías. Los consumidores realizan más de 10 millones de descargas de aplicaciones en los primeros tres días de la App Store y 60 millones en el primer mes”; con esto queda más que claro el gran éxito de esta tienda en línea. Con el paso de los años App Store fue sumando la cantidad de aplicaciones, un año más tarde (2009), esta tienda ya contaba con 50, 000 apps y tal ha sido su crecimiento que en 2013 el registro de apps llego a las 800, 000.

En los primeros meses del lanzamiento del iPhone el impacto aún no era visible en los fabricantes de teléfonos actuales como Motorola, Nokia, Sony Ericsson etc., pero en algún momento tendría que suceder, la tecnología ya los había dejado atrás y es así como comienzan una lucha por ser líderes en la industria móvil. La primer compañía que se pone al tu por tu con Apple es Google quien con su sistema operativo Android ha logrado convertirse en su acérrimo rival todos estos años.

HISTORIA ANDROID

La historia de Android no fue iniciada por Google como muchos piensan, sus raíces se encuentran en una pequeña compañía con sede en Silicon Valley mejor conocida como Android Inc en donde se desarrollo Android, un sistema operativo basado en Linux orientado a dispositivos móviles, como teléfonos inteligentes y tablets; sus fundadores son provenientes de distintas compañías de tecnología en Internet entre los que destacan: Andy Rubin (co-fundador de Danger), Rich Miner (co-fundador de Wildfire Communications), Nick Sears (vicepresidente de T-Mobile) y Chris White (diseñador y desarrollador de la interfaz de WebTV).



HTC DREAM LANZADO EN 2008

Es hasta 2005 que Google adquiere Android Inc, con la idea de demostrar que un sistema operativo móvil podría ser creado y mantenido en una plataforma abierta, contrario a lo que Apple ofrecía. De esta forma Google ayuda a crear el sistema operativo móvil Android 1.0 que sería lanzado en 2008 a través del smartphone HTC Dream; este nuevo sistema genero inmediatamente gran impacto por una parte por tener el respaldo de Google ya que las aplicaciones móviles son alojadas en la Android Market la cual es controlada por dicha compañía lo que es sinónimo de calidad. Por otra parte nace como una alternativa más barata que la competencia con precios más accesibles para la mayor parte de la población. Estos datos fueron consultados en el artículo The History of Android escrito por Sarah Kessler (2011) editora asociada de la revista Fast Company, donde se refiere a las empresas de tecnología, y Felker D. (2011).

Tras ver el éxito de Apple y Google las demás compañías dedicadas a la telefonía celular emprenden su camino hacia los smartphones; Sony Ericsson, Samsung y LG deciden aliarse con Android, Nokia saca su propio sistema operativo el Symbian Os que actualmente ha desaparecido, Black Berry de igual forma lanza su sistema operativo que lleva el mismo nombre y por último esta el sistema Windows Phone creado por Microsoft que ha ido alojándose en distintas marcas de smartphones. Esta competencia ha ayudado a desarrollar nuevas tecnologías, buscar innovar año tras año, mejorar en hardware y software con tal de ser los líderes en la industria móvil.

1.2 ¿QUÉ ES UNA APLICACIÓN MÓVIL?

Para entender que es una aplicación móvil primero se debe conocer que es un smartphone ya que es el medio o el objeto que permite que se ejecutan estas aplicaciones. Uno es dependiente del otro, sin aplicaciones móviles no se podría considerar un teléfono como smartphone y viceversa un teléfono común no podría ejecutar las aplicaciones móviles; de hecho uno nace a partir del otro.

DEFINICIÓN DE SMARTPHONE

Se llama smartphone a cualquier teléfono móvil que cuente con sistema operativo capaz de gestionar un mayor número de datos y ejecutar actividades similares a las de una computadora gracias a las distintas aplicaciones que posee; en cuanto a hardware por lo general cuentan con pantalla sensible al tacto, cámara fotográfica y video, memoria externa, acelerómetro y magnetómetro. Con el uso de smartphones es más fácil y rápido realizar distintas tareas al mismo tiempo por ejemplo agendar reuniones, redactar un mail, saber donde comer, así como mantenerse en contacto con la familia y amigos a través de las distintas redes sociales. Lo importante de este invento es que se ha quitado el apego a estar frente a una PC y lo difícil que es trasladarla de un lugar a otro, ahora en nuestro bolsillo tenemos lo esencial para atender un día laboral o disfrutar del tiempo libre.

Para distinguir un smartphone de un teléfono, Singh S. (2013) Director de Sistemas de GWT Pvt, en su nota “Que son exactamente las aplicaciones móviles” ha sugerido las siguientes características que cualquier smartphone debe tener:

- Poseer un sistema operativo móvil (OS) reconocido, tales como: Android de Google, iOS de Apple, o Windows Phone de Microsoft.
- Conectividad a Internet por medio de Wi-Fi o plan de datos.
- Contar con un navegador móvil.
- Posibilidad de sincronizar varias cuentas de correo electrónico en el dispositivo.
- Hardware y/o software con teclado en distribución QWERTY.
- Sincronización inalámbrica con otros dispositivos, tales como ordenador portátil o equipos de escritorio.
- Posibilidad de descargar aplicaciones y ejecutarlas de forma independiente.
- Capacidad de ejecutar múltiples aplicaciones al mismo tiempo
- GPS (sistema de posicionamiento global).



DIFERENTES MARCAS DE SMARTPHONES

DEFINICIÓN DE APLICACIÓN MÓVIL

Una aplicación móvil es un pequeño software diseñado para ser ejecutado en smartphones, tablets y distintos dispositivos móviles; se refieren a ellas así porque no se pueden ejecutar programas tan complejos debido a que los dispositivos móviles no cuentan con

un disco duro o procesador tan grande que logre soportar demasiado contenido, más bien se podría decir que sus funciones son muy específicas.

Una característica importante que diferencia este software del que comúnmente conocemos, es que estos programas no se instalan por medio de paquetes de CD o algún dispositivo extraíble, están disponibles a través de plataformas de distribución de aplicaciones (sitios web), que son operados por el propietario del sistema operativo para móviles, por ejemplo App Store de Apple, Google Play para Android, Windows Phone Marketplace y BlackBerry App World. Estas aplicaciones son directamente descargadas desde el dispositivo móvil aunque algunas tiendas ya permiten descárgarlas desde cualquier computadora para después ser transferida al dispositivo; algunas aplicaciones son gratuitas y otras de pago, esto va a depender de quien las desarrolle y sin importar si fue comprada o no, las actualizaciones que vengan a futuro serán totalmente gratuitas.

Cuando se habla de este tipo de software es más común escuchar el termino “app” que aplicación móvil a pesar de ser la traducción a nuestro idioma, esta palabra se ha hecho tan popular que en 2010 fue catalogada como palabra del año por la American Dialect Society. En este documento se usaran ambos términos, app y aplicación móvil.

1.3 TIPOS DE APPS

Las aplicaciones móviles se pueden clasificar en dos tipos diferentes en función del modo de desarrollo utilizado para crearlas, estos son: aplicaciones móviles nativas y aplicaciones móviles web. En ambos tipos se pueden obtener resultados muy similares, pero cada una tiene sus ventajas y desventajas dependiendo de cual sea la aplicación a desarrollar.

APLICACIONES MÓVILES NATIVAS

Las apps nativas son programas que se han desarrollado para una plataforma o dispositivo en particular, esto permite que puedan interactuar y aprovechar las características de hardware y software de cada equipo como puede ser la cámara, acelerómetro, GPS, distintos sensores digitales, etc. Estas aplicaciones se instalan directamente en el dispositivo móvil ya sea que estén almacenadas desde la fabricación de este o sean descargadas desde la tienda de aplicaciones correspondiente a su sistema operativo móvil. Los datos asociados con la aplicación también son almacenados dentro del dispositivo aunque en algunos casos pueden ser almacenados en la nube y ser ejecutados en cualquier otro dispositivo.

El hecho de que una app nativa sea desarrollada para una plataforma no significa que sea exclusiva de ella, como nos recomienda Rouse M. (2013) escritora para WhatIs.com: “se pueden hacer distintas versiones para cada plataforma que se desee, siempre tomando en cuenta el lenguaje de programación que corresponda y los cambios que ello conlleva tanto en funciones como en diseño, esto con el fin de permitir una más amplia base de usuarios que accedan a la aplicación;

a este tipo de apps suele llamarse multiplataforma”.

Un ejemplo claro de aplicación móvil nativa son las apps de realidad aumentada particularmente de geo-localización; las cuales aprovechan en primera instancia la cámara del dispositivo con el fin de combinar la realidad con la realidad virtual, la brújula servirá para saber hacia donde está apuntando el smartphone, después por medio del GPS se localizará la ubicación del usuario lo que ayudará a determinar la información que le es

útil en ese momento para ser desplegada en la pantalla, por decir los restaurantes, hoteles o farmacias cercanos y como llegar a ellos, la información que se despliegue será mostrada de acuerdo a la inclinación del dispositivo lo que es determinado por el acelerómetro. En este ejemplo la app está vinculada con el hardware del dispositivo por medio de la cámara, la brújula, el GPS y el acelerómetro; además de estar vinculada con el software del smartphone a través de la aplicación de mapas.



COMPARACIÓN APP NATIVA CON WEB APP

APLICACIONES MÓVILES WEB

En palabras de Singh S. (2013) las web apps son “programas de software que se ejecutan directamente desde el navegador web de los smartphones y tablets; estas aplicaciones están basadas en la web es decir no se instalan en el dispositivo y se ejecutan en los servidores web alojados, este tipo de aplicaciones móviles se pueden desarrollar en una sola plataforma, ya que no están limitados por los sistemas operativos del dispositivo, siendo esta una de sus principales ventajas”.

Algunos ejemplos de webapps son las redes sociales como Twitter y Facebook se accede a ellas mediante el navegador, estas versiones existen por lo lento que sería procesar la gran cantidad de imágenes y videos que manejan. Del mismo modo los periódicos están haciendo lo propio por ser bastantes las secciones que contienen, incluso limitan el número de notas, por ejemplo El Universal o Reforma.

En otras palabras una web app no es más que una página web optimizada para su perfecta visualización en cualquier dispositivo móvil a través de HTML5 y CSS3; a esto se debe de recurrir porque hay sitios muy pesados o con demasiados menús poco visibles o seleccionables para las pantallas de los smartphones o tablets; lo que se pretende con estas web apps es generar una vista más ligera y una fácil interacción con el usuario. Algunas veces su interfaz es tan buena que logra dar la apariencia de una aplicación nativa.



TWITTER Y FACEBOOK

Otra de las ventajas que poseen es que no pasan por procesos de validación de ninguna tienda de aplicaciones pues no son descargables, esto ayuda a que los desarrolladores no tengan que pagar el impuesto revolucionario por las ventas de su aplicación móvil; además de esto el usuario nunca tendrá que actualizar la app ya que al ser una página web, siempre se accederá a la versión más actualizada.

Los contras que tiene para Doncel M. (2012) ingeniero informático, Co-fundador de start-capps son: “el potencial que pierde en marketing al no aparecer en ninguna tienda de aplicaciones; siempre será necesario contar con una conexión a Internet para que pueda funcionar y por último por más que de el aspecto de una aplicación nativa gran parte de las funciones del smartphone no podrán ser aprovechadas al ser el navegador quien ejecute la aplicación y no el dispositivo”.

APLICACIONES HÍBRIDAS

Se conoce como app híbrida a aquella que combina elementos de aplicaciones nativas y aplicaciones Web; sus principales características son la integración con el sistema de archivos de un dispositivo, puede funcionar si el dispositivo está conectado a Internet, integración con los servicios basados en la Web debido a que posee un navegador para mejor el acceso al contenido en línea dinámico.



MINION RUSH

Los ejemplos más claros de estas aplicaciones móviles están en la categoría de Juegos; en ocasiones por los gráficos que contienen pesan demasiados megabytes, lo que demoraría la descarga de la aplicación además de saturar la memoria del dispositivo, es por eso que al abrir la app regularmente se descargan todos esos datos requeridos por el juego desde el navegador, sin la conexión a Internet no hay manera de que funcione la aplicación y suele cerrarse inmediatamente, la app Minion Rush es uno de estos casos. Por otra parte existen algunos juegos que tienen una apariencia totalmente nativa pero que por su modelo

de negocios in app purchase requieren de conexión a Internet para la venta de productos; este modelo de negocios consiste en vender dentro de la aplicación bloques de niveles, personajes, dinero virtual o alguna herramienta que facilite o mejore el juego, como la forma de pago se realiza con tarjeta de crédito o transferencia bancaria se utiliza la ayuda de un navegador en el que puedas ingresar la información necesaria para la transacción.

Existe una gran discusión entre que tipo de aplicaciones móviles son mejores, ya sean mencionado sus principales características y con ello se podría saber cuales son las mejores; no se trata de hacer menos a una u a otro porque todo va a depender de las necesidades del desarrollador, lo que busca transmitir y la forma en quiere llegar al usuario. Sin lugar a duda las app nativas destacan por la interacción con el hardware del dispositivo móvil y el acceso a la información de este, además al no requerir conexión a Internet la batería del smartphone dura por más tiempo. Sin embargo el único aspecto que le podría restar puntos es las veces que debe ser escrito el código de programación de la aplicación para que esta funcione en las diferentes plataformas móviles, además de que la interfaz se ve afectada por el tipo de navegación y las resoluciones de pantalla características de cada sistema operativo móvil haciendo de esto un proceso aún más laborioso.

El hecho de generar una aplicación multiplataforma desde el principio atrae a los desarrolladores hacia las apps Web, pero no es algo que ayude a los usuarios a decidirse por una aplicación, a ellos no les importa si existe la versión en una plataforma que no usa; lo que el usuario quiere es identificarse con una aplicación que se vea como el resto de las aplicaciones en su plataforma, que siga los mismos paradigmas de interfaz que el resto de las aplicaciones que están acostumbrados, y que sea integra en su plataforma.

1.4 PLATAFORMAS DE DESARROLLO

En informática, “una plataforma de desarrollo es el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones. Comúnmente se encuentra relacionada directamente a un sistema operativo; sin embargo, también es posible encontrarla ligada a una familia de lenguajes de programación o a una Interfaz de Programación de Aplicaciones (API por sus siglas en inglés)”. Esta definición fue tomada de Wikipedia (2008).

ANDROID

Es una plataforma móvil basada en Linux (núcleo libre de Sistema Operativo), en su mayoría con un sistema operativo de código abierto, que cuenta con el apoyo de distintos fabricantes de dispositivos móviles. El ser de código abierto ha permitido que los programadores puedan leer y modificar el código fuente, éste evoluciona se desarrolla y mejora a una velocidad mayor a la aplicada en software cerrado permitiéndole de esta forma



LOGO DE ANDROID, JAVA Y GOOGLE PLAY

sobresalir sobre otras plataformas, logrando entrar en la industria móvil como la mejora alternativa de dispositivos de bajo costo, ante los iPhones ya bien establecidos en el mercado. El lenguaje de desarrollo utilizado en este sistema es Java, este lenguaje es una de las principales características que definen a este sistema debido a que cada plataforma cuenta con un lenguaje propio.

Para Ratabouil S. (2012) consultor de IT experimentado con tecnologías Java y C + + “ lo que busca esta plataforma es la portabilidad entre hardware y la capacidad de adaptación a los recursos limitados

de los teléfonos inteligentes; con este Sistema Operativo Móvil, se tiene que hacer frente a múltiples resoluciones de pantalla, varias CPU (Unidad Central de Proceso) y la velocidad o capacidad de GPU (Unidad de Procesamiento Gráfico), las limitaciones de memoria, todo esto a causa de los diferentes modelos que ofrecen al mercado las distintas empresas fabricantes” (p.2).

Algunos datos que identifican y siguen con la filosofía de ser una plataforma de código abierto son las facilidades para ingresar aplicaciones móviles a Google Play (Tienda de apps de Android); en primer lugar la licencia de desarrollador tiene un costo de 25 dólares y es válida de por vida, en segundo lugar el proceso de validación de aplicaciones es bastante flexible tiene una duración de 5 a 30 minutos.

Con todas estas opciones disponibles se pueden llegar a hacer grandes cosas y ser capaces de llegar al mercado mundial de una manera fácil, pero ¿qué tiene de atractivo esta plataforma además de su libre desarrollo, por qué debemos decidarnos por ella? La respuesta que Ostrander J. (2012) desarrollador Web y de software móvil, tiene “es la gran expansión que ha tenido rápidamente a nivel mundial sobre todo en los Estados Unidos en donde Android se ejecuta en la mayoría de los teléfonos inteligentes en sus diferentes tipos. Su expansión va más allá de los smartphones, recientemente ha entrado al mercado de las tablets con una línea amplia de productos por ejemplo la Nexus 7. Es rápido ganando terreno en el mundo del desarrollo, especialmente en África, donde la necesidad de teléfonos de bajo costo es mayor” (p. VIII).

A través de Google Play es como se pueden conseguir las apps para dicha plataforma que en su mayoría son gratuitas pero también existen de pago, para las cuales existen dos formas de pago: una por cargo en tarjeta de crédito y la otra por medio de Google Cards aunque aún no ésta disponible en México.

Los peros que se le pueden poner a la plataforma es que la mayoría de las aplicaciones son de mala calidad visualmente hablando, poco podrían competir con apps de iOS; hacen falta procesos de validación que no solo se enfoquen en el funcionamiento de está, también es necesario evaluar la usabilidad y la experiencia de usuario. Hablando de este proceso de validación la flexibilidad con la que cuenta la ha hecho el blanco perfecto para malware que en este caso son aplicaciones móviles que sirven para infectar el dispositivo con distintos tipos de virus, estas aplicaciones suelen ser disfrazadas y por ellos no son detectadas.

iOS

iOS es una plataforma móvil desarrollada por la empresa Apple, desde su creación ha sido un referente de la industria móvil por su calidad, fluidez y diseño impecable mostrado en sus diferentes apps. El lenguaje utilizado para poder desarrollar en esta plataforma es Objective-C a través del entorno de desarrollo formado por el SDK (kit de desarrollo de software) y Xcode. Ambos entornos son proporcionados por Apple de manera gratuita pero solo pueden ser utilizados en computadoras de escritorios o portátiles Macintosh.

El objetivo de está plataforma es crear aplicaciones móviles centradas en lo que el usuario quiere hacer en determinado momento, simples pero no por su contenido sino por su manera de usar, es decir deben ser fáciles de usar; si de algo se preocupa este sistema operativo es de la usabilidad y lograr la mejor experiencia de usuario posible desde los desarrolladores hasta los usuarios finales. A partir del SDK Apple se preocupa por brindar un entorno fácil de comprender para el desarrollador; en la experiencia de Goldstein M.(2013) líder en la fabricación de tecnologías de vanguardia: “con un equipo de trabajo conformado por dos desarrolladores y un buen diseñador gráfico se puede hacer una aplicación móvil de muy buena calidad” (p.16).



LOGO DE iOS, APP STORE Y OBJECTIVE C

Apple ha sido sinónimo de innovación durante muchos años, sus productos han servido de agente de cambio en distintas industrias; el iPhone fue capaz de revolucionar la industria móvil, por su parte el iPad está cambiando por completo el Internet como un medio de publicación, la industria del software se ha visto beneficiada en las distintas apps lanzadas. Esta empresa sirve de ejemplo para muchas otras es por ello que se preocupa por la imagen que da a los demás, sus estándares son de lo más estricto en todos sus productos tanto en equipos como en software; para que una aplicación móvil pueda ser vendida en la App Store debe de pasar por un proceso de validación que va de una semana a quince días, es más tardado que otras plataformas porque las pruebas van más allá de su funcionamiento, se cuida que se sigan los principios de interfaz de usuario fundamentales y la forma en que son aplicados. No aceptan aplicaciones hechas de un día para otro pues se les hace una falta de respeto hacia los desarrolladores serios y profesionales.

La App Store es una tienda muy bien manejada y cuidada, en ella solo podemos encontrar las mejores aplicaciones móviles, por tal motivo la licencia para desarrollar en iOS no la puede obtener cualquiera, su costo es de 99 dólares por año, demasiado caro en comparación con la competencia Android. Si bien este podría ser un punto en contra, la forma en que Apple gestiona cada app hace que valga completamente pagar este cargo; anunciará la aplicación en la categoría que el desarrollador especifique, hará el procesamiento de tarjetas de crédito si es que la aplicación es de pago, hosting, descarga e informar a los usuarios de las actualizaciones son solo algunas de las tareas que realizara por el desarrollador.

La forma de adquirir apps en esta tienda es muy variada y se adecua a los distintos bolsillos de la gente; en primera están las aplicaciones gratuitas sin ningún tipo de cargo por su descarga, en este rubro podrían entrar aplicaciones de promoción, es decir apps de pago que por alguna razón tienen algún descuento o por algunas semanas son gratuitas. En segundo lugar están las apps de pago, aquí existen dos formas de pago, una es por medio de tarjeta de crédito haciendo el cargo a través de una cuenta bancaria y la otra forma es por medio de tarjetas de iTunes las cuales se puedes conseguir en supermercado o tiendas departamentales, estas tarjetas cuentan con cierto crédito que servirá para descargar las apps. Cabe señalar que en las aplicaciones móviles de pago, Apple recibe el treinta por ciento del costo de éstas; sin duda es una herramienta de productividad de negocios.

En este modelo de negocios los expertos de la compañía hablan de tres diferentes estilos de aplicación para desarrollar, Goldstein M. (2013) hace mención de ellos en iOS dummies estos son:

- Aplicaciones de productividad utilizan y manipulan la información. Aplicaciones de iWork de Apple: Keynote, Pages y Numbers. Común a todas estas aplicaciones es el uso y la manipulación de los múltiples tipos de información.
- Aplicaciones de utilidad realizan tareas simples, muy definidas. YouTube de

Google la aplicación es un ejemplo que sólo se refiere a los vídeos de YouTube. La aplicación Sketchbook que sirve para dibujar se considera una utilidad, ya que realiza una tarea simple, altamente definida.

- Aplicaciones inmersivas. Se centran en entregar y tener al usuario interactuando con contenidos en un entorno de gran riqueza visual. Un juego es un ejemplo típico de una aplicación inmersiva, como el caso de Angry Birds con su variedad de personajes y escenarios que contiene cada nivel.

Es importante señalar que estos estilos son muy parecidos a las categorías de la Apps Store pero estas son solo una designación de marketing. También es necesario aclarar que una app desarrollada en iOS no forzosamente es exclusiva de este Sistema Operativo, puede existir su versión en Android y viceversa, aunque llegan a darse casos de apps únicas para iOS.

METODOLOGÍA

DEL DISEÑO

CAPÍTULO 2

2.1 METODOLOGÍA

Se cree que a la hora de diseñar existen dos tipos de enfoques, uno es el creativo que ve el proceso mágico donde ocurre la misteriosa chispa creativa, no existe una explicación lógica de cómo surgen las ideas, se dan por sí solas; el otro es el enfoque racional, que ve al diseño como producto de procesos racionales completamente explicables, y que deben ser definidos a partir de una estructura ordenada. Siguiendo este último enfoque se ve la necesidad de crear alguna metodología que ayude a llegar al objetivo de la manera más fácil y eficiente, por tal motivo primero se debe conocer que es el método, la metodología, por qué son necesarios y que métodos existen debido a que en cada proyecto se podría utilizar uno diferente, dependiendo del problema a solucionar.

Para comenzar el método se define como el camino que conduce al conocimiento; por su etimología griega “metha” a través de, “odos” camino. Es decir la serie de pasos que se utilizan para llegar a tal resultado. La metodología se refiere a la ciencia que estudia los métodos del conocimiento; expone la manera en que se organiza el proceso de investigación, el control que se tiene sobre los resultados y la toma de decisiones ante las posibles soluciones a algún problema. Se encarga de analizar y criticar los diferentes métodos de investigación.

En el ámbito del diseño diferentes autores han dado su definición a estos términos enfocándose a dicha disciplina, siempre tomando en cuenta el método que ellos proponen y las ideas que defienden.

Luz del Carmen Vilchis (2002) define la metodología del diseño como: “el conjunto de disciplinas en las que lo fundamental es la concepción y el desarrollo de proyectos que permitan prever cómo tendrán que ser las cosas e idear los instrumentos adecuados para los objetivos preestablecidos” (p.41).

En contraparte Cross (1996) ve la metodología como el “estudio, análisis, de la manera de trabajar y pensar del diseñador, el establecimiento de las estructuras adecuadas para representar el proceso de diseño así como el desarrollo y aplicación de nuevos métodos y técnicas, aunada a la reflexión sobre la naturaleza y alcance del conocimiento con objeto de aplicarla a problemas de diseño” (p. 56).

El método es considerado por Christopher Jones (1976) como el “medio para resolver el conflicto entre el análisis racional y el pensamiento creativo”. Papanek (1984) lo considera como “la interacción de herramientas, tratamientos y materiales que se usaran óptima, económica y eficientemente en el proyecto de diseño”. En resumen el método representa un número variado de actividades que el diseñador emplea dentro del proceso de diseño en el que logra combinar la parte teórica (conceptos, definiciones, elementos, fundamentos, etc.) con su creatividad. Será el método quien logre comprobar lo factible que son

las ideas surgidas por el pensamiento creativo, a través de experiencias estudiadas o por lógica de su contenido.

Las metodologías de diseño parten de los tipos de métodos que existen para cualquier materia o disciplina, por tal motivo la clasificación hecha por Bochenski (1969) resulta de gran ayuda para entender de donde parten dichas metodologías, en dicha clasificación se mencionan:

MÉTODOS DEL PENSAMIENTO PRÁCTICO

Que se ordena hacia algo que puede hacer quien piensa, que se emplea cuando se quiere lograr un saber, pero un saber cómo, esto es, cómo se puede hacer esto o aquello.

MÉTODOS DEL PENSAMIENTO TEÓRICO

Que se ordena a los contenidos que pudieran aprehenderse independientemente de su utilidad práctica.

MÉTODO DEDUCTIVO

Método clásico del razonamiento a priori. Es aquel que parte de datos generales aceptados como válidos y que, por medio del razonamiento lógico, pueden deducirse varias suposiciones. Se suele decir que se pasa de lo general a lo particular.

MÉTODO INDUCTIVO

Se caracteriza por establecer los conceptos a partir de los hechos, por inferir lo general de lo particular, a diferencia de la deducción que establece verdades necesarias, llega a verdades probables.

MÉTODO INTUITIVO

Este método se divide en cuatro fases:

1ª La evidencia o intuición: No admitir nada como verdadero, nada más que lo evidente, claro y distinto. La intuición se define como “El movimiento del espíritu puro y atento capaz de captar ideas evidentes, claras y distintas”. La fuente de este proceso de conocimiento inmediato es la razón cuyo objetivo es establecer una idea verdadera que sirva para fundamentar lo científico.

2ª El análisis: Hay que dividir las dificultades encontradas en cuantas partes sea posible y necesarias para su resolución.

3ª La síntesis: Conducir los pensamientos con orden, empezando por lo más simple y fácil de conocer, e ir ascendiendo gradualmente hasta el conocimiento de lo más compuesto, suponiendo un orden entre ellos.

4ª La revisión: Hacer recuentos y revisiones completas y generales de modo que se esté seguro de que no se omite nada en el proceso.

MÉTODO DIALÉCTICO

Es una actitud con respecto al objeto, empírica y deductiva, e impone con ello cierta forma de recoger datos concretos, representa una forma tentativa de explicación de los hechos sociales. Se basa sobre todo en los conocimientos que se aprenden a través de la experiencia de cada individuo, esto ayuda a decidir lo que es correcto y lo que no (p. 29).

2.2 METODOLOGÍAS DEL DISEÑO

CHRISTOPHER JONES

Pionero en introducir el método al diseño tras observar que era una necesidad, sus postulados se basan en los conceptos de caja negra y caja transparente. El primero Jones (1976) lo considera: “una teoría que valora de una manera importante el subconsciente; la parte más valiosa del proceso del diseño se realiza en el subconsciente del diseñador. Difícilmente sometido a las leyes objetivas de la ciencia, no puede limitarse a la camisa de fuerza de la sistematicidad, ya que cada problema es solucionado de manera distinta por cada diseñador, no existe una explicación clara de cómo se llegó a tal resultado, por lo tanto es muy difícil encontrar la forma de repetirlo”.

En el concepto de caja transparente la razón será el factor a considerar acompañada de la planificación. Este proceso se divide en tres etapas: Análisis, Síntesis y Evaluación.

En el **análisis** se hace una investigación sobre todos los factores que están conectados al problema, para identificar lo que podría cambiar y lo que sería una constante, de este modo se logra un panorama más amplio y fructífero para la búsqueda de una solución.

La etapa de **síntesis** trata de combinar los juicios de valor subjetivo con los factores encontrados en la etapa de análisis para lograr un modelo lo suficientemente preciso que tenga como resultado un solo diseño a detallar. Se fijan los objetivos y límites del problema, las condiciones de cada variable y con ello emitir un juicio sobre cual es la más conveniente.

La última de las etapas, **la evaluación**, se concentra en encontrar una única alternativa para llegar a la solución; cuando se llega a esta etapa es porque se está más cerca del diseño total, se ha hecho una reducción progresiva de las incertidumbres secundarias.

Jones (1976) no considera a un método como mejor que el otro, lo que postula es que “con el método racional se tiene la impresión de que puede resolverse el problema de diseño con matemática certeza, pero que, en realidad, lo bueno es combinar racionalidad e intuición, algo mucho más difícil que ser solamente racional o solamente creativo”.

BERND LÖBACH

La metodología propuesta por Löbach tiene como finalidad lograr un producto reproducible tecnológicamente. En su libro Diseño Industrial, Löbach (1981) enumera las siguientes fases como parte de su metodología:

1. Análisis del problema. Punto de partida cuyo planteamiento generalmente es presentado al diseñador por la empresa. Se recopilan todos aquellos datos que le atañen, sin menospreciar detalles; algunos de estos datos son: vínculo entre el problema usuario y el objeto, ambiente en que se insertará el producto, competencia, datos técnicos acerca del uso del objeto, materiales y procesos de fabricación.

2. Soluciones al problema. También conocido como brief, será la elección ordenada, estratégica y creativa de los datos que permitirán definir los objetivos de forma correcta. Con base en las relaciones de la información y la conclusión de condiciones para la solución del problema, el diseñador incursiona en la fase propiamente creativa. Se definen diversas posibilidades para resolver el problema en cuestión, a través de bocetos o modelos de prueba de las soluciones pensadas.

3. Valoración de las soluciones del problema. Aquí tiene lugar el examen minucioso de las alternativas presentadas entre las que se elige aquella que responde a un enfrentamiento cuidadoso con los valores fijados como conclusiones de la fase 1.

4. Realización de la solución del problema. En ella se concreta la respuesta y afinan los mínimos detalles con dibujos y explicaciones gráficas necesarias.

GUI BONSIPE

La metodología propuesta por Bonsipe (1978) es a mi parecer una combinación de las metodologías de Löbach y Asimow del cual se hablará un poco más adelante; lo que aporta en su metodología será una serie de técnicas específicas o recomendaciones que facilitarán algunos pasos del proceso, éstas se refieren a :

Análisis funcional, para describir la función de un objeto, sus componentes o interacciones.

Análisis morfológico, como técnica combinatoria para formular posibles conjuntos de soluciones para un mismo problema.

Sinéctica, consiste en el rastreo para localizar otras posibles soluciones para un problema, a partir de acciones como analogía, inversión, amplificación, sustitución, empatía, etc.

Síntesis formal, que indica los aspectos perceptivos del objeto: forma, color, textura, etc.

Optimizar características de uso mediante la ubicación de criterios con respecto al usuario: ergonomía, seguridad, simplicidad de uso, limpieza, mantenimiento, acceso, etc.

Visualizar ideas básicas empleando códigos visuales: dibujos, esquemas, grafos, maquetas, prototipos.

Como se ha visto las metodologías del diseño suelen ser muy parecidas entre ellas, Vilchis (2002) y Frascara (1998) hacen mención de las constantes metodológicas éstas son:

1. Encargo del trabajo por el cliente

2. Información e investigación: consiste en el acopio y ordenamiento del material relativo al caso o al problema particular.

3. Análisis: Es la descomposición de un todo en sus elementos creando con ello prioridades y jerarquías.

4. Síntesis: consiste en la propuesta de criterios válidos para la mayor parte de demandas y que el conjunto se manifieste en un todo estructurado y coherente.

5. Evaluación: concerniente en la sustentación de la respuesta formal a la contrastación con la realidad. Se presenta el producto al cliente y éste dará el visto bueno

6. Organización de la producción: Requerimientos para la producción del producto.

7. Implementación: introducción del producto en el mercado.

8. Verificación: se trata de medir el éxito o fracaso del producto y con ello evaluar la continuidad o el retiro del mismo.

BRUNO MUNARI

Llamado método proyectual, se basa en una serie de operaciones necesarias, colocadas en orden coherente dictado por la experiencia. La principal característica es la subdivisión de problemas, pues Munari (1981) postula que en estos “subproblemas se encuentra la solución al problema mayor, con esta partición es más fácil encontrar los elementos que forman el todo” (p. 352). Las fases que lo componen son:

Definición del problema. Hay que detallar el problema, ubicarlo en contexto, hacia quien va dirigido.

Definición y reconocimiento de subproblemas. Son los pequeños problemas particulares, que resolviendo uno a uno, permiten conocerlos mejor y finalmente integrarse en una pieza o conjunto de piezas de diseño.

Recopilación de datos. Documentarse sobre el tema al cual vamos a trabajar, de lo que ya existe sobre éste tema aplicado gráficamente a un producto de diseño, y como se ha resuelto en otras ocasiones.

Análisis de datos. Sacar una conclusión en base a lo investigado, tanto de la competencia como de la temática.

Creatividad. Lluvia de ideas, relaciones de palabras con formas y colores, lecturas, etc.

Materiales. Recursos que serán necesarios para diseñar como pueden ser lápices, pinturas, pinceles, gises, papel.

Tecnología. En la actualidad estos recursos son muy importantes pues reducen los tiempos de producción, se trata de analizar si es mejor utilizar una cámara fotográfica digital o analógica, dibujar en una tableta o en papel.

Experimentación. Básicamente es bocetar, jugar con todos los recursos que se tienen, las ideas que van surgiendo plasmarlas en papel o en algún otro medio,

Modelos. Serán lo más acercado a la solución final, existe un trabajo más elaborado.

Verificación. Se presentan las ideas a los posibles usuarios del diseño para su juicio sincero. Se realizan encuestas o alguna forma de evaluación para obtener información valiosa para la retroalimentación.

Solución final. Se hacen los ajustes requeridos de acuerdo a la evaluación anteriormente realizada para obtener una solución definitiva.

MORRIS ASIMOW

La metodología que propone Asimow esta basada en el diseño, la producción y distribución de algún producto, desde los primeros pasos se hace hincapié en si es viable en estos aspectos, denotando que no se trata de diseñar y ya después encontrar la manera de vender la propuesta.

Asimow (1962) parte de la idea de “encontrar una necesidad (sensación de carencia unida al deseo de satisfacerla), la cual el diseño ayudará a satisfacer; identificando la necesidad, continua el razonamiento sobre si es un producto físicamente realizable y económicamente costeable” (p. 12).

Después el diseño toma mayor peso dentro del proceso; se comienza con un diseño preliminar donde se descubren aspectos como el estilo gráfico, apariencias, etc., es el bocetaje

de las ideas que se tienen, se trata de desarrollar distintos conceptos para posteriormente elegir el más óptimo. Ya que es elegido, será momento de realizar un diseño detallado, en el que se hace una subdivisión de las tareas a elaborar; para entender mejor cada una de ellas y como será la metodología utilizada para esta tesis pondré como ejemplo el diseño de una aplicación móvil, estas tareas consisten en:

Preparación del diseño. En esta etapa se hace una división de las pantallas que formarán la aplicación (splash screen, menú, ajustes, información, etc.)

Diseño total de los subsistemas. Cada una de estas pantallas requiere de distintos elementos gráficos como puede ser logotipos, íconos, imágenes, botones, etc., se debe identificar cual de ellos debe llevar cada pantalla.

Diseño total de los componentes. Consiste en una visión general de la pantalla, la interacción que habrá entre elementos y su colocación.

Diseño detallado de las partes. Como su nombre lo indica es la elaboración de cada elemento gráfico a profundidad, por decir un ícono de que color será, su tamaño, que va a indicar, si estará acompañado de algún texto.

Preparación de los dibujos de ensamble. Es la forma de interactuar entre pantallas, la manera en que las partes sea verán como un todo y creen una composición, que sea vea un estilo definido.

Construcción experimental. Se podría decir que es un dummy, lo más parecido al producto final con el que se podrán hacer las pruebas pertinentes.

Programa de pruebas del producto. Es la evaluación del producto por parte del usuario no final o de algún experto para conocer los errores y aciertos que se tiene, probar si es funcional el producto y en el caso de este ejemplo son importantes los conceptos de usabilidad y experiencia de usuario que más adelante serán estudiados

Análisis y predicción. En base a los resultados obtenidos en las pruebas, analizar que debe cambiarse y que cosas podrían mejorarse.

Rediseño. Hacer los ajustes correspondientes al diseño con respecto al análisis previamente elaborado. A veces es necesario cambiar por completo el concepto por lo que las tareas de esta fase pueden volverse un ciclo.

Las últimas fases corresponden a un nivel administrativo y de negocios que no por eso debe descuidar el diseñador o no intervenir, en ellas se planificará la **producción del producto**, la forma de **distribución** (plan de distribución) y como el **cliente lo podrá adquirir** (plan de consumo). Cuando esté ya se encuentre en el mercado evaluar la productividad y las ganancias que ha dejado por si es conveniente **dejarlo o retirarlo** (plan de retiro).

Estás dos últimas metodologías, Munary y Asimow, a mi parecer son las que mejor se adaptan para el desarrollo de una aplicación ya que, como veremos en el capítulo 4 de ésta tesis, se necesita de planeación desde la concepción de la idea, organizar la información recabada sobre la investigación de nuestros usuarios, a partir de ella crear posibles soluciones que serán probadas a través de modelos o prototipos parciales o totales, para posteriormente someterse a evaluación, comprobar que cumplen con su objetivo y sobre todo si son un producto viable que satisfacen la necesidad de los futuros usuarios. Con esta subdivisión de tareas que proponen estos autores, existe gran similitud con las metodologías utilizadas para la elaboración de software las cuales también serán tratadas en el cuarto capítulo.

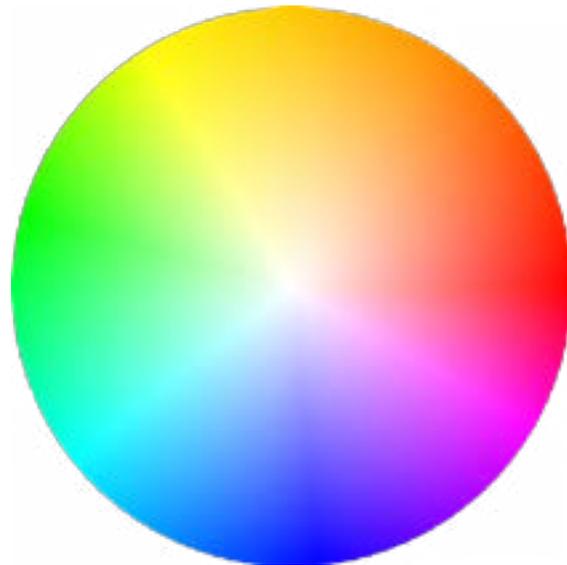
GRÁFICOS

PARA APPS

CAPÍTULO 3

3.1 COLOR

Es una propiedad física de la luz emitida por los objetos y sustancias; percepción visual generada en el cerebro de los humanos al interpretar las señales nerviosas que le envían los fotorreceptores en la retina del ojo, que a su vez interpretan y distinguen las distintas longitudes de onda que captan de la parte visible del espectro electromagnético.



MODELO DE COLOR RGB

MODELO RGB

En los smartphones el color es emitido por la luz de sus pantallas por lo tanto el color se muestra basado en el modelo aditivo de color RGB (rojo, verde, azul, en sus siglas en inglés), se denomina aditivo al modelo porque la combinación de estos colores en estado puro produce la luz blanca. Aunque todos los dispositivos se basen en este modelo, no quiere decir que todos muestren el mismo rojo, verde o azul; la configuración de la pantalla como puede ser el brillo logra afectar la percepción de los colores, además de la resolución de la pantalla y la cantidad de memoria destinada a la tarjeta de video que se ocupa de controlar el color de cada píxel, regularmente se destina 8 bytes. Cuando se reservan 8 bits de memoria, cada píxel puede mostrar uno de entre 256 colores (256= 2 elevado a la octava; en otras palabras, 256 es el máximo número de combinaciones diferentes de ceros y unos que se pueden obtener con 8 bits de memoria).

Para indicar con qué proporción es mezclado cada color, se asigna un valor a cada uno de los colores primarios, de manera que el valor 0 significa que no interviene en la mezcla y, a medida que ese valor aumenta, se entiende que aporta más intensidad a la mezcla. Así, de manera usual, la intensidad de cada una de las componentes se mide según una escala que va del 0 al 255 y cada color es definido por el valor que se le da a R, G y B. Cuando el valor de los tres colores es 255 forman el color blanco mientras que, cuando el valor es 0 para los tres colores el resultado es negro.

MODELO HEXADECIMAL

Este modelo es más utilizado en la Web y en diversos tipos de programación por la facilidad con la que se escribe el valor de cada color; este modelo utiliza la combinación de tres códigos de dos dígitos para representar las intensidades de cada uno de los colores

primarios RGB. Es decir al color R (rojo) le corresponden dos cifras que representarán la intensidad que aporte al color final, esta cifra va del 0 al 9 y además se utilizan seis letras con un valor numérico equivalente, a=10, b=11, c=12, d=13, e=14, y f=15. Lo mismo pasa con G (verde) y B (azul) con lo que se formarán los tres códigos del color final. Por ejemplo con el amarillo el código #ffff00 el rojo y verde están al máximo y el azul no aporta intensidad en este color.

El porque se utilizan letras y de acuerdo con Wikipedia (2013), tiene que ver con que en principio, dado que el sistema usual de numeración es de base decimal, sólo se dispone de diez dígitos, por ello se adoptó la convención de usar las seis primeras letras del alfabeto latino para suplir los dígitos que hacen falta.

La correspondencia entre la numeración hexadecimal y la decimal o RGB viene dada por la siguiente fórmula:

decimal = primera cifra hexadecimal * 16 + segunda cifra hexadecimal

La intensidad máxima es ff, que se corresponde con $(15*16)+15= 255$ en decimal, y la nula es 00, también 0 en decimal. De esta manera, cualquier color queda definido por tres pares de dígitos.

Los tres colores básicos:

Rojo #ff0000 El canal de rojo está al máximo y los otros dos al mínimo

Verde #00ff00 El canal del verde está al máximo y los otros dos al mínimo

Azul #0000ff El canal del azul está al máximo y los otros dos al mínimo

Las combinaciones básicas:

Amarillo #ffff00 Los canales rojo y verde están al máximo

Cian #00ffff Los canales azul y verde están al máximo

Magenta #ff00ff Los canales rojo y azul están al máximo



COLORES UTILIZADOS EN LA VERSION 7 DE iOS

EN INTERFAZ

El color es un recurso vital en el diseño de una aplicación. Su uso abarca encabezados, textos, botones, fondos y muchos otros elementos que conforman la interfaz.

En algunas ocasiones, está asociado a la identidad corporativa y en otras, responde a criterios estéticos y decisiones de diseño.

Los colores utilizados en las plataformas iOS y Android son cambiados con relativa frecuencia dependiendo de las actualizaciones de su sistema, algunas tempora-

das son colores pasteles o neutros, en otras son muy vibrantes sin embargo su propósito sigue siendo el mismo, crear una interfaz de usuario capaz de mejorar la experiencia de usuario y facilitar el manejo de la aplicación móvil; el color ayuda a la interactividad, da vida a la interfaz y genera continuidad visual. No porque estas plataformas utilicen ciertos colores significa que todas las apps deben usar los mismos colores, es válido proponer, crear colores personalizados asegurando que funcionan bien juntos, se debe crear una paleta de colores que se puedan utilizar en toda la aplicación. Estos dos sistemas suelen utilizar los mismos colores solo que en el caso de iOS se aplica con degradados y algunas transparencias; mientras que en el caso de Android suele ser utilizado en plasta o plano.

En la guía de desarrolladores de iOS se menciona que es importante el prestar atención en los contrastes de color en diferentes contextos, por ejemplo si no hay suficiente contraste entre el fondo y la barra de navegación de los títulos de barras de botones, los botones resultaran difíciles de observar por el usuario. Se menciona en esta guía como regla de oro el comenzar con un contraste de al menos 50% entre los colores que deben ser distinguibles. El mejor consejo que se da en esta guía en cuanto a color y para descubrir las áreas que necesitan mayor contraste es desaturar la interfaz de usuario y mirarlo en escala de grises. Si existe problema para diferenciar entre elementos o fondos interactivos y no interactivos en la versión de escala de grises, es recomendable aumentar el contraste entre estos elementos.

Como en cualquier soporte en que se aplica el color, este tendrá algún significado y causara algún efecto en el usuario, siempre y cuando tenga un uso consistente, consciente y vinculado al contexto donde se aplica, no por ser colores transmitidos desde una pantalla perderán su valor o significado.

EN TEXTOS

Ya mencionado anteriormente el color además de dar identidad a la app, es útil en cada elemento de la interfaz de usuario, llámese texto o fondo por poner un ejemplo. En el caso del



texto, el color puede ser usado para jerarquizar el contenido, la información importante puede ser destacada con un color diferente de el resto del contenido y al contrario la información secundaria puede estar en tonos más claros sin descuidar la legibilidad.

Aunque es más visto en la web, el color también se puede usar para destacar enunciados o palabras que servirán como enlaces, para ello es relevante permitir que el usuario intuya y a simple vista logre distinguir los elementos tipográficos interactivos.

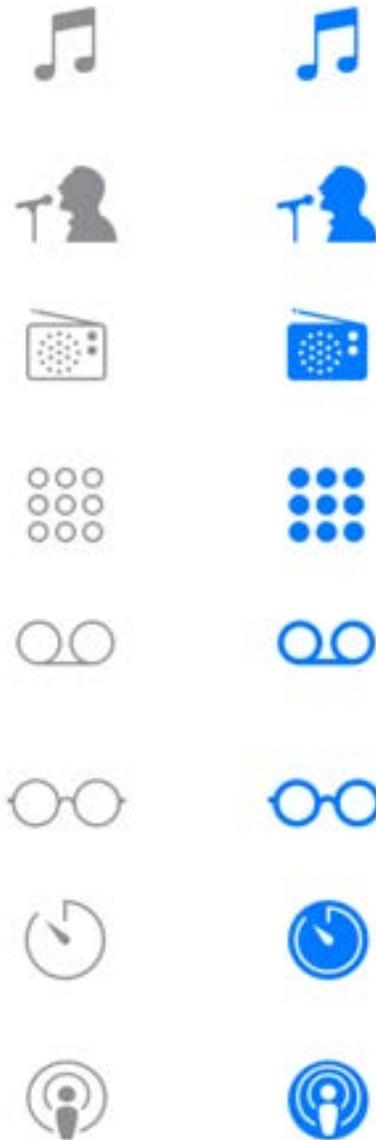
EN FONDOS

Para la elección del color de fondo hay que tomar en cuenta el contraste con la tipografía y con los elementos interactivos, debido a que factores como la accesibilidad y la legibilidad se pueden ver afectados.

De igual forma que en el papel, y como lo menciona Monjo (2011) profesor y master universitario en Ingeniería informática, “los colores oscuros cansan la vista más rápidamente, por lo tanto si la app es de uso frecuente o se requiere pasar un largo periodo de tiempo leyendo, es conveniente revisar la elección cromática y buscar alternativas más claras de color”. Esto no quiere decir que se descarten los colores oscuros, pueden ser utilizados para dar mayor énfasis en elementos visuales como fotografías y videos o en menús laterales que contrasten perfectamente con la pantalla principal ayudando a distinguir entre contenidos y opciones.

EN ELEMENTOS INTERACTIVOS

En la mayoría de las aplicaciones móviles el color es quien muestra el cambio de estado de un botón y con ello la ejecución de alguna acción que se le este ordenando al smartphone,



CAMBIO DE ESTADO EN ICONOS

esto le permite al usuario saber que la aplicación está reaccionando o que se está llevando a cabo algún proceso. Los elementos seleccionados o pulsados, como botones o filas, pueden destacarse con un color que indique visualmente dónde se ha pulsado, lo cual suele ser particularmente difícil de saber en un móvil.

Al elegir el color clave para la interactividad y el estado es recomendable que ese color sea único y exclusivamente utilizado para eso, ya que si es usado en elementos interactivos y no interactivos será más difícil para el usuario identificar donde debe tocar, asimismo se debe asegurar que los demás colores no compitan con el para reconocer fácilmente estos elementos interactivos.

Cuando los elementos están deshabilitados el color puede ser más claro que en su estado activo o puede usarse el mismo color pero con transparencias u opacidad distinta, incluso en la nueva versión de iOS suele emplearse el mismo color pero en estado desactivado el elemento es mostrado en contorno y en estado activo ya se muestra con relleno. De cualquier forma, el objetivo es indicar de una forma evidente que ese elemento producirá algún efecto al ser pulsado.

Vittone (2012) diseñador de interfaces en Usolab, cree que hay algunos colores que ya están destinados para mostrar interactividad, incluso los llama colores reservados, en su libro dice que: “hay ciertos colores que deben emplearse de forma cuidadosa porque tienen connotaciones que no pueden obviarse. Se llaman justamente «colores reservados» porque su uso debería limitarse especialmente a los nombrados a continuación:

- **Rojo:** Para errores y alertas importantes. Es un color que naturalmente indica peligro y llama la atención para centrarse inmediatamente en lo que está ocurriendo.
- **Amarillo:** Prevención. Señala que la acción que va a realizarse implica la toma de una decisión que ocasiona alguna consecuencia, por lo cual hay que estar alerta.
- **Verde:** Mensajes de éxito y confirmación de que una acción se ha realizado correctamente.”

Sin embargo considero que esto puede confundir porque da a entender que todos los botones deberían ser verdes y si ocurre algún problema estos deberían transformarse a rojo; a mi parecer el color va a depender mucho de la identidad de la aplicación, por ejemplo si se trata de una app de Coca-Cola no vas a dejar de usar el color rojo tan característico de esta empresa solo porque indica alerta.

3.2 TIPOGRAFÍA

La tipografía es uno de los elementos que componen la interfaz gráfica de las apps, la cual se refiere al equilibrio y la correcta interacción entre los distintos caracteres en pantalla; es tan importante como cualquier otro elemento visual que se incorpore en una interfaz

Arial
Futura
Impact
Myriad
Tahoma

BONDI
Century
Garamond
Georga
Times New Roman

TIPOGRAFÍAS SERIF Y SANS SERIF

pequeña, la tipografía sobre todo la vemos en títulos, menús, resultados, pero estos siempre son muy breves, solo en casos de algunas aplicaciones que muestran instrucciones de uso es que existe una mayor cantidad de caracteres pero sin exagerar el contenido.

El uso de mayúsculas o de minúsculas afecta en gran medida a la percepción de un texto sean párrafos o títulos. De hecho deben evitarse las palabras en mayúsculas exceptuando tal vez los títulos cortos, pues su rastreo es más difícil. Al leer no visualizamos cada una de las letras que componen una palabra y luego la construimos, sino que, en primer lugar, reconocemos la forma de la palabra como un conjunto de letras. Esta forma de escritura es mucho más cómoda y legible pues, al leer lo primero que visualizamos es la parte superior de las palabras.

La buena legibilidad que una tipografía puede tener en papel puede diferir de forma importante al ser usada en pantalla, esto se debe a que algunas fuentes son más legibles que otras en pantalla. El ejemplo que Ferrer A. (2011) Licenciada en Bellas Artes, menciona en su libro Escritura y tipografía es que “una fuente tradicional como puede ser la Times New Roman, considerada una de las más legibles en papel, no puede tener la misma consideración en pantalla; lo que ocurre es que en cuerpo pequeño, sus formas se ven irregulares y pierde legibilidad. La legibilidad en la pantalla depende sobre todo de la altura de la

y su elección no debe descuidarse. A su vez juega dos papeles importantes, visual y verbal, el primero porque el usuario observa a primera instancia la composición del todo y con esto las letras únicamente forman parte de la imagen, y a segunda instancia es verbal porque decodifica el lenguaje de cada palabra y lee la información que se le esta mostrando.

LEGIBILIDAD

Como en cualquier diseño, el objetivo de la tipografía es conseguir que el texto se lea con claridad, es decir, sea legible. Esto se logra no solo con una adecuada elección de la fuente, sino también gestionando su tamaño, separación entre líneas, ancho de columnas y contraste visual con el fondo.

El uso de bloques de texto en las apps es poco recomendable ya que es muy difícil la lectura en dispositivos de pantalla pe-

“x” definida en las tipografías, mientras más alta mejor y la geometría del tipo evitando una presencia excesiva de curvas” (p. 125).

Para tener una mejor legibilidad en la pantalla y no abandonar las tipografías más comunes se les hacen adaptaciones, aunque también se da el caso de crear nuevas fuentes específicamente para pantallas de distintos dispositivos. Éstas últimas están hechas para leerse con mayor comodidad en pantalla, por tal motivo tienen una exagerada altura de la “x” y son más robustas que las tipografías para papel.



VISUALIZACIÓN DE TIPOGRAFÍAS

Aunque su uso está más ligado a la web, las llamadas web-safe-fonts son un buen parámetro para la elección de tipografía para móviles, se pueden encontrar muchas y diferentes tipografías actualmente consideradas fuentes óptimas para mostrarse en pantalla, son de todo tipo y de familias diferentes; son llamadas así porque pueden ser visualizadas en cualquier navegador de Internet y tienen mejor legibilidad, además de que el usuario ya está acostumbrado a ellas. Algunas de las más utilizadas son:

Serif

Bookman Old Style, Garamond, Georgia, Palatino, Linotype, Book Antiqua y Times New Roman.

Sans-serif

Arial, Arial Black, Helvética, Gadget, Impact, Charcoal, Geneva, New York, Trebuchet, Verdana, Lucida sans unicote y Tahoma.

Serif o sans-serif

Nuevamente en diseñando apps para móviles, Vittone (2012) hace referencia acerca de la discusión que se tiene para elegir una fuente; esta elección recae o suele concentrarse entre las tipografías con serifa y las que no. Si bien es cierto que en tamaños pequeños y bajas resoluciones rinden mejor aquellas fuentes más limpias, abiertas y sans-serif, también pueden considerarse las serif para los títulos principales, cuando cuenten con un tamaño en el que las serifas no sean un impedimento para la lectura. Desde mi punto de vista lo más

Text Size Micro	12sp
Text Size Small	14sp
Text Size Medium	18sp
Text Size Large	22sp

recomendable son las tipografías sans serif por algo Apple y Android usan tipografías de este tipo para su sistema, son fáciles de leer y su lectura es muy fluida. Las tipografías serif suelen ser utilizadas en aplicaciones móviles de arte o historia, pero como dice Vittone en tamaños grandes.

REPRESENTACIÓN DE LOS TAMAÑOS MÍNIMOS

TAMAÑOS MÍNIMOS

En dispositivos móviles el tamaño de la tipografía estará limitado de acuerdo al tamaño del dispositivo, si se trata de smartphone este suele ser más pequeño que la tablet por lo tanto la distancia a la que los sostenemos es más cercana a los ojos del usuario permitiendo que el tamaño de la tipografía sea pequeño, obviamente sin excederse. Caso contrario son las tablets las cuales buscan una mayor comodidad por parte del usuario, estas pueden apoyarse en algún soporte y se miran a una mayor distancia, por lo tanto los gráficos son más grandes y por supuesto la tipografía no puede dejar de serlo.

De acuerdo a la guía para desarrolladores de Android, el tamaño tipográfico se mide en sp (píxeles escalados), es llamado así porque es una forma de modificar la escala de las fuentes de acuerdo al tamaño de pantalla y a las preferencias definidas por el usuario en su configuración del teléfono. Los tamaños más comunes van desde 12sp hasta 22sp.

Por su parte en la guía de iOS, el tamaño es variable dependiendo de dónde está ubicado el texto. En el caso de la densidad «retina», los títulos principales rondan los 34px y el tamaño en etiquetas dentro de botones importantes es aproximadamente 28px. A partir de allí, va disminuyendo en los diferentes elementos hasta llegar al tamaño más pequeño, cercano a los 14px. Sin embargo, se recomienda no usar dimensiones inferiores a 20px en los textos de lectura.

JERARQUÍAS

La tipografía en la interfaz tiene diferente valor de acuerdo a su función y a la información que contiene; esto permite crear jerarquías con las que el usuario logra identificar los aspectos más importantes.

Las características con las que se distinguen las jerarquías de texto dentro de una pantalla pueden ser el tamaño, las variantes negrita, bold, regular o light y el color. Lo más común es que el nombre de la pantalla sea el tamaño más grande a utilizar en toda la pantalla y de ahí ir disminuyendo el tamaño de acuerdo a su jerarquía.

En el caso del color en la tipografía y como se mencionaba anteriormente siempre debe estar en contraste con el fondo para permitir la mejor legibilidad posible. El contraste, es más importante de lo que puede creerse a simple vista. Al ojo le resulta atractivo un contraste bien aplicado para de esta manera distinguir un elemento de

otro. También hay que pensar el ambiente en que será usada la app, puede que sea usada el mayor tiempo en exteriores y con esto el sol dará directamente sobre la pantalla y si no hay un buen contraste entre tipografía y fondo, la información en pantalla será imposible de leer.

TIPOGRAFÍA ANDROID

Android y iOS tienen una tipografía por default para su sistema, así como un set de fuentes que pueden ser utilizados por los diseñadores de su respectiva plataforma. El elegir alguna de las opciones disponibles lo ayudará con su usuario en el reconocimiento del sistema en que ha sido desarrollada la aplicación móvil.

De acuerdo con Cuello (2013) diseñador de aplicaciones móviles de Zara y Yahoo, y la guía de desarrolladores de Android, la familia Droid Sans fue de las más utilizadas y marcó una época dentro de la dicha plataforma. Sin embargo, en las últimas versiones de este sistema operativo ha sido reemplazada por Roboto, especialmente diseñada para móviles de alta resolución y con variantes que van desde una extra delgada hasta black.

Roboto Regular

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Roboto Bold

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

TIPOGRAFÍA ROBOTO

Neue Helvética Light

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

Neue Helvética Medium

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

TIPOGRAFÍA HELVETICA NEUE

TIPOGRAFÍA iOS

Por su parte en iOS y consultándolo en su guía de desarrollo, la fuente emblemática de este sistema en los últimos años ha sido Helvética Neue y es normal encontrar muchas aplicaciones que la usan. Con el paso del tiempo se ha transformado casi en una marca de identidad de este sistema operativo; no obstante, hay más de 260 fuentes disponibles que pueden usarse de forma nativa.

Cada uno de estos sistemas operativos también permite incorporar en el diseño fuentes adicionales a las predefinidas. Aunque esto sea posible, es importante recordar que variedad y calidad son dos cosas diferentes: muchas de las fuentes que pueden encontrarse, no han sido pensadas ni preparadas especialmente para una buena legibilidad en pantalla.

Aunque esto tenga que ver más con programación, podría ser una preocupación para el diseñador si cualquier tipografía podrá ser utilizada como cualquiera propuesta por el sistema, pero no hay que preocuparse la solución a esto está dentro de la guía de desarrolladores de cualquiera de las dos plataformas, al momento de programar la aplicación cuando se adopta un estilo dinámico, se obtiene:

- Ajustes automáticos de espaciado entre letras y altura de la línea para cada tamaño de la fuente
- La capacidad de especificar diferentes estilos de texto para los bloques semánticamente distintos de texto, como el cuerpo, la nota, o Headline
- El texto que responde de forma adecuada a los cambios que el usuario realiza los ajustes de texto de tamaño (incluyendo tamaños de texto de accesibilidad).

En general, lo mejor es utilizar una sola fuente a través de su aplicación. Mezclando varias fuentes diferentes puede hacer que la aplicación parezca fragmentada y descuidada. En su lugar, conviene utilizar una fuente con algunos estilos y pocos tamaños.

3.3 RESOLUCIONES DE PANTALLA

La resolución de pantalla se refiere al número de píxeles que puede ser mostrado en la pantalla. Un píxel (px) es la unidad de medida más pequeña en una pantalla y representa un cuadrado físico minúsculo, en ocasiones es posible ver estos cuadritos sobre todo cuando se dice que una imagen está pixeleada.

Por ser la unidad más pequeña no es posible tener una línea menor de 1 píxel de ancho, es imposible que exista una línea de 0.4 píxeles el software de gráficos lo interpreta como una línea de 1 píxel más débil.

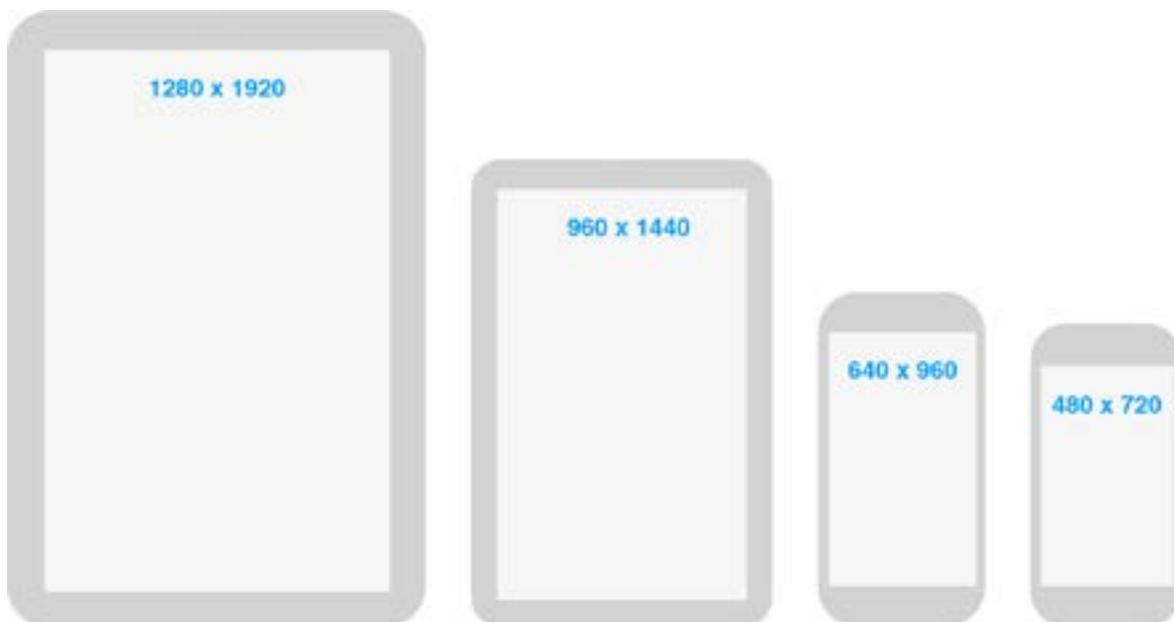
La resolución de cada dispositivo móvil va a variar dependiendo del tamaño y el tipo de pantalla con el que cuenta, como es el caso de la pantalla retina exclusiva de los dispositivos Apple las cuales serán descritas más adelante. Antes de pasar a las diferencias de pantallas de acuerdo al Sistema Operativo, un punto importante a considerar es que las imágenes no son las únicas afectadas por la resolución, la tipografía también debe de ajustarse de acuerdo a la resolución del dispositivo, que si bien no se verá borrosa o pixeleada puede perder legibilidad.

PANTALLAS ANDROID

En el caso de los dispositivos Android existe una gran variedad de modelos por lo que no hay estándares de resolución. Lo que esta plataforma hace es clasificar los dispositivos en pequeños, medianos, largos y extralargo.

Los pequeños son aquellos que no exceden los 600 píxeles de ancho; los medianos son aquellos que no exceden los 1 200 píxeles de ancho. Aquellos que no exceden los 2 500 píxeles de ancho son considerados largos y por último los extralargos se refieren a pantallas similares a un televisor.

Para simplificar la forma de diseñar para múltiples pantallas, lo que se hace es partir del tamaño medio equivalente a 1 y de ahí partir hacia las otras clasificaciones; pequeño es equivalente a .75 del medio, largo a 1.5 del medio y extralargo al doble del medio. Como ejemplo y para que sea más claro, diseñando la pantalla de splash screen tomamos la resolución de 640 x 960 píxeles equivalente a la media, ahora para dispositivos pequeños se necesita un splash de 480 x 720 píxeles que equivale al 75 % de la media. Para los dispositivos largos será necesaria una imagen de 960 x 1440 píxeles y para dispositivos extralargos de 1280 x 1920.



RESOLUCIONES DE PANTALLA ANDROID

Todo esto es necesario para optimizar la interfaz de usuario y permitir que cada elemento se escale de acuerdo a la resolución del dispositivo y aunque suene un poco tedioso este proceso, existen programas que son capaces de transformar las imágenes o piezas a cada tamaño simplemente proporcionando el tamaño más grande que será utilizado.

PANTALLAS iOS

Caso contrario son las resoluciones de Apple debido a que no existen tantos productos con iOS únicamente se trata de iPhone, iPod, y iPad; si bien hay distintas resoluciones, los cambios que se presentan son por el tamaño del dispositivo y por el cambio de generación.

En el caso del iPhone desde su aparición y hasta la cuarta generación tenía una resolución de 340 x 480 pixeles, fue con la aparición de la versión 4S que cuenta con pantalla retina que hubo un cambio en la resolución al duplicarse pasando a 640 x 960. Con la quinta generación el producto se volvió más alto y mantuvo la pantalla retina cambiando su resolución a 640 x 1136 px.

La tablet de Apple contó en sus inicios con una resolución de 768 x 1024 px, al ser de los últimos productos lanzados por la compañía únicamente duró una generación sin pantalla retina, para las generaciones posteriores se convirtió en resolución de 1536 x 2048 px. Para mayor comodidad y mejor manejo por parte del usuario se creo a finales de 2012 la iPad mini con un tamaño más pequeño pero con una resolución igual a la de iPad 768 x 1024 px y 1536 x 2048 px en versión retina. Aquí es importante aclarar que el tamaño del dispositivo a veces no esta en relación con la resolución, esto se debe a la densidad de pixeles por pulgada de la pantalla la cual determina la nitidez de ésta.



RESOLUCIONES DE PANTALLA iOS

Lo que diferencia una pantalla normal de una retina de acuerdo con Taylor (2013), es que esta última “es una pantalla con alta resolución con la intención de hacer píxeles invisibles, esto quiere decir que la retina es un duplicado exacto de la resolución normal; para ello un único píxel en un dispositivo móvil iOS no retina es absorbido por cuatro píxeles en la versión retina, esto hace más nítido el texto y permite mucho más detalles sutiles.”

3.4 ÍCONOS

LANZADOR

El icono de lanzador es la representación visual de la aplicación en la pantalla de inicio, aunque ocupa una pequeña parte de ella debe proporcionar al usuario una idea de lo que realiza la app, es su tarjeta de presentación. En la mayoría de los casos, las personas basan sus opiniones iniciales sobre la calidad, efectividad y fiabilidad de la aplicación únicamente con el aspecto de su icono. Al diseñar el ícono lo más recomendable es seguir estos tres principios:

1. Sencillez. Para conseguir un buen icono, lo más importante es la sencillez. Como en la mayoría de los casos en diseño gráfico, menos es más. El icono de una aplicación sirve para representar la esencia de ésta y debe hacerlo estéticamente con gráficos simples e informativos. Por su tamaño, es muy fácil que un icono sea excesivo, por lo que es importante limitar los colores, estampados, textos e imágenes al mínimo.

2. Usar colores con moderación. En el caso de los colores, utilizarlos de forma inteligente es imprescindible para tener un icono exitoso. Un icono lleno de colores vibrantes abrumará al receptor. Deben evitarse los estampados y limitar la paleta de colores a uno o dos. Además, es imprescindible respetar los colores de la aplicación en sí y mantener los mismos tonos.

3. Cuanto menos texto, mejor. El mundo móvil tiene muy poca tolerancia al texto por lo que no es necesario utilizarlo en el icono de una aplicación. Una simple imagen, con la marca será suficiente para representar el servicio. En el caso de que sí requiera texto, debe limitarse solo a una línea, intentando mantener siempre la claridad del icono, son muy pocas las veces en que se ocupa ya que todas las plataformas agregan por default en la pantalla de menú el nombre de la app.

Al igual que en las resoluciones de pantalla, es importante tomar en cuenta la variedad de dispositivos y pantallas en las que el ícono se puede visualizar para que en todas ellas se



ÍCONOS DE APPS



ÍCONOS EN APP STORE Y GOOGLE PLAY

Lo que distingue a estos iconos es que pueden estar sin fondo, ya que se permiten las transparencias.

En el caso de dispositivos iOS el ícono del lanzador es de 120 píxeles para iPhone 5S y iPod touch de 5a generación, para iPad y mini iPad es de 152 píxeles en su versión retina y 76 píxeles para la versión estándar. El tamaño para la App Store debe ser de 1024 x 1024 píxeles. Su principal característica es que están dentro de una envolvente cuadrada con esquinas redondeadas.

Cuando la aplicación se ejecuta, el sistema operativo comprueba las características de la pantalla del dispositivo y carga los archivos de tamaño específico apropiados para su visualización. Dado que el usuario puede cambiar el papel tapiz de la pantalla de inicio en cualquiera de las dos plataformas, hay que comprobar que el ícono de lanzador es claramente visible en cualquier tipo de fondo.

INTERIORES

En el interior de la aplicación se encuentran los íconos que servirán como herramientas de trabajo o navegación y que su papel es más funcional que estético aunque no por eso deben descuidarse. Pueden llegar a ser desapercibidos pero su trabajo es comunicar a la aplicación móvil con el usuario.

Los íconos interiores son utilizados principalmente en tres casos: el primero de ellos como ayuda para reforzar cierta información de manera visual, por ejemplo cuando aparece algún diálogo preguntando si se quiere abandonar la aplicación. El segundo caso se refiere cuando actúa como elemento interactivo, el ícono se convierte en botón o pestaña; en el último caso, ayuda a optimizar el espacio al resumir de manera visual textos muy extensos o difíciles de comprender.

Con base en la guía de desarrollo de Android, se recomienda que al diseñar algún ícono de este tipo se respete el tamaño mínimo de 48 píxeles. Tomar en cuenta este tamaño

y respetar este criterio de dimensiones para los botones, asegura que estos podrán ser tocados con cualquier dedo de la mano sin problemas, cuestión fundamental para la usabilidad de la app. Al juntarse varios íconos interiores se crea una barra de herramientas en la que deben de ser tomados en cuenta el espaciado y la separación, la recomendación de tamaño que se da para este módulo es de 8 píxeles mínimo. En los márgenes laterales se suele usar una retícula con 16 píxeles de separación.



ÍCONOS INTERIORES VERSIÓN IOS 7

Las mismas recomendaciones deben de tomarse en cuenta para iOS con la diferencia que el tamaño mínimo base es de 44 píxeles, asegurando que los botones o elementos interactivos puedan ser pulsados sin problemas. Los espacios y separaciones son un poco más grandes en comparación de Android, su tamaño es de 11 píxeles mínimos, generando un ritmo vertical

Existen diversos tipos de iconos interiores: herramientas, navegación, pestañas, etc., por tal motivo cada uno tiene sus propios tamaños mínimos, al final de este capítulo se muestra la tabla de tamaños de íconos e imágenes de la guía de desarrollo de iOS.

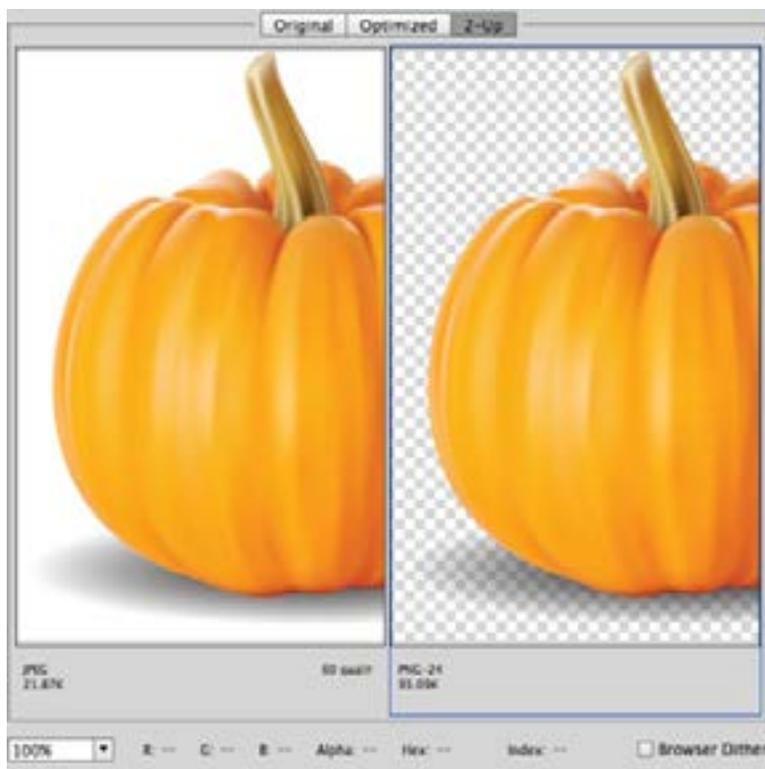
3.5 FORMATO DE IMAGEN

PNG

El formato de imagen más utilizado en dispositivos móviles es el PNG (Portable network Graphics), debido a su ligero tamaño por la compresión de imagen, lo que permite que el flujo de la aplicación sea continuo.

De acuerdo con Monjo (2011) diseñadora en experiencia de usuario, “este formato permite obtener archivos de imagen muy comprimidos a diferentes niveles de calidad; genera imágenes a 8 o 24 bits dependiendo de la resolución de pantalla de cada dispositivo. Su mejor característica es permitir transparencia en los píxeles con 65 niveles disponibles por imagen, lo que permite transparencias graduales, sombras, desenfoques” (p. 290).

Cada imagen en este caso en formato .png representa una pieza individual en la interfaz de usuario. Por ejemplo, los botones serán nombrados botonmenu.png y el fondo de la apli-



COMPARACIÓN FORMATO JPG Y PNG

JPEG

Muy pocas veces utilizado pero si compatible con dispositivos móviles es el formato de archivo JPEG (Joint Photographic Experts Group), también considerado un método de compresión; es el más utilizado por dispositivos de captura de imagen digital, además sirve para el almacenamiento y la transmisión de imágenes fotográficas en la Web. Consultando en Wikipedia, el formato JPEG utiliza un algoritmo de compresión con pérdida para reducir el tamaño de los archivos de imágenes; esto significa que al descomprimir o visualizar la imagen no se obtiene exactamente la misma imagen de la que se partía antes de la compresión. El algoritmo de compresión JPEG se basa en dos fenómenos visuales del ojo humano: uno es que capta más claramente los cambios de brillo que de color. El otro efecto es que nota con más facilidad pequeños cambios de brillo en zonas homogéneas que en zonas donde la variación es grande.

Una de sus principales características es la maleabilidad con la que se ajusta el grado de compresión aunque está presenta sus inconvenientes. Si hay un grado de compresión muy alto se generará un archivo de tamaño pequeño, pero con pérdida significativa de calidad. Ocurre lo contrario ocurre con un grado de compresión bajo, en el que se obtiene una calidad de imagen muy parecida a la original, sin embargo su tamaño de archivo se hace mayor. La desventaja que tiene contra el formato PNG es que no permite transparencia y el fondo suele ser de color.

cación `fondotienda.png`. Absolutamente todas las piezas o imágenes que se le den al diseñador deben de estar en este formato.

En el caso de la resolución retina se deben crear dos imágenes para cada pieza, una de ellas en resolución normal o 1x y la otra en retina al doble de la normal o 2x; todas las piezas retina deben ser exactamente dos veces más grande en cada dimensión que las piezas en normal, por ejemplo un botón de 48 x 65 píxeles tendrá en resolución retina un tamaño de 96 x 130 píxeles.

Con el entendimiento y comprensión de estos elementos gráficos más las guías de diseño de cada sistema operativo, es como se puede lograr un buen diseño visual que de seguimiento a la arquitectura de la información, le de estilo e identidad a nuestra app, además de ser atractiva para el usuario. Pero no solo es ello, también es un elemento importante para guiar a los usuarios dentro de la aplicación móvil y que le ayude a completar sus tareas, creando un mejor flujo visual que se refiere a la rapidez y facilidad con que el diseño visual transmite la información y con ello una buena experiencia de usuario.

En el siguiente capítulo nos daremos cuenta como interviene el diseño gráfico con los elementos presentados en este capítulo, además de los conocimientos de estructuración y organización con los que cuenta un diseñador que servirán para constituir los diferentes contenidos que pueden tener las apps.

Tabla de tamaños de íconos en píxeles e imágenes personalizadas.

Descripción	Tamaño para el iPhone 5 y el iPod touch (de alta resolución)	Tamaño para el iPhone y el iPod touch (de alta resolución)	Tamaño para el iPad y el iPad mini (alta resolución)	Tamaño para el iPad 2 y el iPad mini (resolución estandar)
Icono de la aplicación (se requiere para todas las aplicaciones)	120 x 120	120 x 120	152 x 152	76 x 76
Icono de la aplicación para la App Store (requerida para todas las aplicaciones)	1024 x 1024	1024 x 1024	1024 x 1024	1024 x 1024
Imagen Launch (requerida para todas las aplicaciones)	640 x 1136	640 x 960	1536 x 2048 (vertical) 2048 x 1535 (paisaje)	768 x 1024 (vertical) 1024 x 768 (paisaje)
Resultados de búsqueda de Spotlight icono (recomendado)	80 x 80	80 x 80	80 x 80	40 x 40
Icono Configuración (recomendado)	58 x 58	58 x 58	58 x 58	29 x 29
Barra de herramientas y barra de íconos de navegación (opcional)	Cerca de 44 x 44	Cerca de 44 x 44	Cerca de 44 x 44	Alrededor de un 22 x 22
Icono de la barra de pestañas (opcional)	Cerca de 50 x 50 (máximo: 96 x 64)	Cerca de 50 x 50 (máximo: 96 x 64)	Cerca de 50 x 50 (máximo: 96 x 64)	Cerca de 25 x 25 (máximo: 48 x 32)
Icono de portada predeterminada Newsstand para la App Store (requerido para aplicaciones Newsstand)	Al menos 1024 píxeles en el lado más largo	Al menos 1024 píxeles en el lado más largo	Al menos 1024 píxeles en el lado más largo	Al menos 512 píxeles en el lado más largo
Clip Webicono (recomendado para aplicaciones web y sitios web)	120 x 120	120 x 120	152 x 152	76 x 76

DESARROLLO
DE
APLICACIONES
MÓVILES

CAPÍTULO 4

La creación de una app implica todo un proceso interdisciplinario en el que se ven involucrados diseñadores, ingenieros desarrolladores, mercadólogos e incluso administradores; este proceso abarca desde la concepción de la idea hasta el análisis de resultados posterior a la publicación del software en las tiendas.

Para darle vida a una aplicación se necesita de un método de desarrollo, actualmente existen varios de donde elegir, cada uno con sus fortalezas y debilidades, lo importante es seleccionar de acuerdo a las necesidades del cliente o al grupo de trabajo. Prieto (2011) hace mención de algunos de los métodos más conocidos: “Modelo waterfall, desarrollo rápido, desarrollo ágil y Mobile-D; estos se basan en la creación de procesos cortos que pueden ser continuos o independientes, los equipos de trabajo suelen ser pequeños y se apoyan de soluciones conocidas o consolidadas. Lo que varía de un método a otro son los tiempos de entrega, el permitir cambios o no sobre la marcha y el considerar un prototipo como un producto final que se mejorará con el tiempo” (p.167).

A partir de estos métodos, se pueden establecer como las fases más comunes:

Idea, Análisis (Requerimientos), Arquitectura de Información, Diseño, Desarrollo, Prototipos y Evaluación.

Antes de explicar cada una de las fases es importante señalar que el desarrollo de una aplicación móvil suele ser de corta duración, debido a causas como los repentinos cambios en las tendencias tecnológicas, las novedades que se presentan por lo menos cada seis meses tanto en software como en hardware, la competencia que existe en el sector o nuevas necesidades de la gente. Esta es la explicación del por qué muchas de la aplicaciones en tienda sobre todo en el caso de Android parecen prototipos que poco a poco irán evolucionando o son aplicaciones muy simples que no necesitan de un gran esfuerzo.

Para comprender mejor este proceso de desarrollo, con ayuda de la app “Pluzer” se logrará ejemplificar cada fase, por ello en primera instancia serán explicadas teóricamente y posteriormente se describirá como fueron adoptadas por la aplicación móvil antes mencionada.

4.1 IDEA

Una aplicación móvil no se crea de la nada, surge a partir de una idea la cual es una propuesta de cómo se puede resolver alguna necesidad; si bien siempre se busca originalidad y tener una app innovadora, las ideas pueden aparecer de algo que ya exista para tratar de mejorarla, superarla o darle otro enfoque.

Hay quienes se quiebran la cabeza tratando de conseguir una idea, pero no se trata de eso, lo único que se necesita es investigar y no sólo se trata de libros o fuentes de consulta

como el Internet, también de algo tan simple como mirar alrededor, observar el entorno, las cosas que hacen los demás, la forma en que interactúan entre ellas, los problemas que tienen en lo que comúnmente hacen, incluso analizándonos a uno mismo puede aparecer una grandiosa idea.

Cuando ya se tiene la idea en mente lo mejor es investigar en Internet o en las tiendas de aplicaciones las apps que podrían ser parecidas a la nuestra; si llega a existir alguna aplicación móvil muy semejante no se trata de desilusionarse y echar todo a la basura, lo mejor es analizar cada una de ellas, observar como resolvieron tal situación o que errores cometieron, se trata de entender cómo pueden ser mejoradas y darle mayor valor a nuestra propuesta. Esta investigación va a permitir enriquecer la idea inicial, marcar diferencia y probablemente tener mayor éxito.

Tal es el caso de Whatsapp, en la App Store se encuentran muchísimas apps destinadas a la mensajería instantánea pero que en su mayoría se tenían que crear perfiles o claves muy extrañas y resultaba complicado encontrar gente conocida; Whatsapp encontró la manera de evitar estas claves tomando el número telefónico del usuario y todos aquellos contactos guardados en su agenda telefónica para que automáticamente formen parte de los contactos de la app. La experiencia de usuario, la simplicidad, las funciones etc. serán quienes marquen la diferencia.

El siguiente paso es comprobar si la idea funcionará o más bien si realmente es útil tal como la planeamos o deba corregirse, creo que no se puede hablar de funcionamiento hasta que no exista mínimo un prototipo. La forma de verificar la idea es preguntar a los posibles usuario su visión acerca del concepto planteado, si les parece buena o mala, qué agregarían o quitarían, si realmente les ayuda con alguna necesidad; también puede llevarse a cabo esta comprobación comentando la idea con familiares, amigos o desconocidos para que den su punta de vista quizás de una manera más informal.



La idea nace a partir de lo tardado o tedioso que suele resultar compartir nuestros datos personales con otra persona, como por ejemplo teléfono, email, Facebook, etc., debe ser uno por uno y por lo regular se tiene que estar verificando que los datos sean correctos; si bien las tarjetas de presentación hacen esta labor, poco a poco el formato impreso ha quedado en el pasado, es momento de aprovechar las nuevas tecnologías sobre todo en los dispositivos móviles.

Pluzer es una aplicación móvil que actúa como una red social en donde el usuario crea tarjetas de presentación con los datos personales que desee y las comparte mediante su representación QR. Éstas tarjetas al momento de ser leídas por la cámara de un smartphone generan un contacto, dichos contactos serán almacenados en forma de agenda.

4.2 ANÁLISIS

El análisis o también llamado requerimientos, consiste en la reunión de información sobre las acciones e interacciones que el usuario necesita para conseguir los objetivos de la app, es decir las funciones que debe tener, así como las características de los usuarios potenciales y los requisitos técnicos y humanos del desarrollo. La forma en que se consiguen estos datos es por medio de la observación del usuario en las posibles situaciones en que necesitará de la aplicación o por medio de entrevistas al cliente donde poco a poco se irán mencionando las características que requiere.

Los primeros requerimientos van enfocados al tipo y contenido de la app, esto es si va a ser una red social, un juego, una enciclopedia, un reproductor de música, un producto promocional, etc.; el contenido va a especificar a que categoría pertenecerá en la tienda de aplicaciones.

Después lo mejor es decidir para qué plataforma desarrollar (iOS, Android, Windows Phone, Black Berry) ya que cada una de ellas podría o no satisfacer por completo los requerimientos; hay funciones que se pueden adaptar a una u otra o puede darse el caso en donde se tenga que eliminar. Así mismo para tomar la decisión de desarrollar una aplicación multiplataforma se debe tomar en cuenta los recursos con que se cuentan, la complejidad del desarrollo, el tipo de usuario al que se quiere llegar y si se tratará de una app nativa, web o híbrida.

De acuerdo a un estudio realizado por Koetsier (2013) vicepresidente de producto, “los sistemas operativos con mayor penetración en el mercado son Android y iOS, en ese orden. Entre ellos se reparten las mayor parte de usuarios con smartphone, mientras el tercer puesto esta ocupado por Windows Phone que poco a poco va a la alza en la industria móvil”.

Como tercer paso esta delimitar el target (tipo de usuario al que va dirigida) porque va a determinar el diseño y la navegación, por un lado puede tratarse de un producto de difusión general donde existe mayor variabilidad de usuarios, con o sin experiencia en las apps, o por otro lado puede dirigirse a un público altamente especializado que ya conoce como funcionan las aplicaciones. Además de la experiencia, la edad del usuario se convierte en factor al priorizar los factores de accesibilidad en niños y adultos mayores sobre público objetivo como el juvenil o adulto.

Para especificar el perfil que tendrá el target no será a base de suposiciones o teorías, se realizan técnicas de modelado de usuario que estarán fundamentadas en el análisis previo sobre la posible audiencia, esto permitirá reunir información sobre sus atributos comunes, necesidades, experiencias y conocimientos, o las condiciones en que acceda a la aplicación móvil.

TÉCNICA DE PERSONAS

Cuando se trata de un público diverso puede resultar difícil definir sus patrones de conducta, objetivos y necesidades; de acuerdo con esta técnica popularizada por Cooper (2011) padre del lenguaje de programación Visual Basic, se utilizan arquetipos llamados personas que son descritas de forma narrativa y se les atribuye una identidad concreta (nombre, fotografía, empleo, edad, etc.). El objetivo de la utilización de personas consiste en que el diseñador tenga en mente a un usuario real con ciertas características determinadas. Todas las características de la persona deben ser coherentes con la información obtenida sobre la audiencia en la fase de análisis.

Mientras más detallada sea la descripción de la persona mejores resultados se obtendrán, las características fundamentales corresponden a:

- **Perfil geográfico.** Lugar de residencia y de trabajo, y nivel de vida.
- **Perfil demográfico.** Edad, género, familia, ingresos, ocupación, educación, etc.
- **Perfil psicosocial.** Clase social, estilo de vida, aficiones, características personales.

Continuando con la técnica de personas, la relación que existe entre el producto (en este caso la app) y la persona delimitará el tipo de usuario, estos pueden ser:

- **Focal.** Es el principal tipo de usuario al que nos dirigimos, y en cualquier proceso de diseño debería existir al menos una persona focal. Se dan casos en que hay más de una persona focal por el rol que puede ocupar en la aplicación, por ejemplo pueden existir administradores que controlen el contenido y que muy probablemente requieran diferente tipo de interfaz al del usuario común.
- **Secundaria.** Los usuarios secundarios también utilizan el producto, y deben usarlo satisfactoriamente, aunque en caso de conflicto de intereses, se priorizarán las necesidades del usuario focal. Es muy común en los juegos al ser pensados para niños que sus gráficos sean sencillos y coloridos, y cuando un adulto llegó a jugarlo le parezca divertido pero pida gráficos más avanzados, los creadores del juego no tendrían que cambiarlos porque su usuario meta es a quien busca complacer.
- **Involucrado.** No utiliza el producto, pero puede verse afectado por los resultados. Los juegos para niños por lo regular cuentan con una sección un tanto oculta para el usuario focal, pero que esta diseñada para que los padres puedan controlar las compras que realizan sus hijos en la aplicación, con la finalidad de prevenir cargos excesivos en las tarjetas de crédito.
- **Promotores.** Pueden generarse minipersonas que correspondan a los intereses de los promotores del proyecto (cliente, inversores, publicistas, etc.), con el objetivo de

garantizar que sus requisitos serán tomados en cuenta. El ejemplo más claro son las páginas de Facebook regularmente controladas por alguna empresa que busca anunciarse por este medio, la interfaz de esta red social cambia completamente respecto al perfil de cualquier usuario, existen mayor números de campos por llenar, más tipos de contenidos e incluso se puede programar el momento en que se haga cierta publicación.

- **No prioritario.** Usuarios infrecuentes, no autorizados o de baja prioridad.
- **Excluido.** Usuario para el que no se está diseñando.

TÉCNICA DE ESCENARIOS

El plantear las situaciones específicas y el contexto en que será utilizada la app permitirá crear escenarios que ambienten el proceso de interacción entre la persona y la aplicación, con ello jerarquizar las tareas que deben llevarse a cabo y comprobar cuales de los requerimientos iniciales siguen siendo útiles y cuales se deben desechar.

La descripción del escenario debe efectuarse utilizando un lenguaje narrativo y directo; por ejemplo Carlos es un doctor que todos los días viaja hacia su trabajo en coche, por el caos que hay en la ciudad debe de buscar vías alternas para llegar a tiempo a su consultorio etc. De igual forma que en las personas, todos los detalles deben ser coherentes con la información obtenida en la fase de análisis.

En el cuarto paso se toma en cuenta el tiempo de vida de la aplicación móvil, porque impactará en el estilo que se le aplique a la interfaz; cuando el tiempo de vida es corto se vale arriesgarse, proponer algo diferente o acoplarse a la moda o corriente estilística actual. Si se trata de tiempo de vida largo lo mejor es ser más conservador y pensar que esa interfaz debe ser buena durante algunos años. Por ejemplo una app que sea pensada para la copa mundial de futbol de Brasil tendrá a lo mucho un periodo de vida de dos meses, contemplando su lanzamiento un mes antes del evento para mayor distribución más el mes que dura la competición, por lo que toda la aplicación debería llevar el estilo que maneja dicho evento para que la gente reconozca que se trata de un producto del mundial. Caso contrario a la app de la FIFA que no debe inclinarse hacia algún evento, su estilo debe conservarse más tiempo porque lo importante es la identidad de la organización.

Cuando ya se tienen los requerimientos técnicos por completo, el último paso será destinar el personal que se necesita para el desarrollo de la aplicación móvil; diseñador y desarrollador deben considerar a las personas adecuadas para cada tarea y los roles que desempeñaran en el proceso. Es importante señalar que los tiempos de entrega y los recursos económicos, afectan la cantidad de personal que se contrate.

Los primeros en contemplar serán el equipo encargado de la gestión de la producción, estos organizarán al personal, verificarán que las tareas se estén llevando a cabo, cuales son los recursos humanos y tecnológicos disponibles, además de establecer las condiciones relativas a recursos económicos y tiempos de desarrollo.

Después entraría la mano de obra, es decir diseñadores, programadores y técnicos que se encargarán de darle vida a la aplicación: estructurando los contenidos, diseñando la interfaz, escribiendo código, realizando pruebas, etc.



Categoría: Red Social

Plataformas: iOS y Android

Técnica de personas:

Persona 1

Nombre: José López Ibarra

Edad: 26 años

Domicilio: La Condesa, Ciudad de México

Ocupación: Periodista

Clase social: Media alta

Características personales: Es un joven soltero extrovertido, muy sociable por lo tanto busca hacer nuevas amistades, le gusta divertirse saliendo a bailar y es fanático de la tecnología.

Persona 2

Nombre: Leticia Gutiérrez

Edad: 38 años

Domicilio: Coyoacán, Ciudad de México

Ocupación: Publicista

Clase social: Media alta

Características personales: Es una mujer casada muy talentosa, que participa en diferentes Expos como conferencista, como resultado de su trabajo necesita de buenas relaciones públicas para generar contactos de negocios, por ello maneja diversas redes

sociales como medio de comunicación que son visualizadas en cada una de sus presentaciones.

Técnica de escenarios:

José es un joven que le gusta salir de fiesta todos los fines de semana; en uno de esos días conoce en el antro a una chica llamada Karen, se le acerca y comienza a platicar con ella, pero el alto volumen de la música no permite una buena conversación. Tras darse cuenta de la dificultad José le pide su número telefónico a Karen para que mejor hablen en otro momento, ella accede y empieza a dictárselo, él apunta el número y verifica que sea correcto pero tiene un fallo en algún dígito por lo que debe de repetir el teléfono, cuando es confirmado ahora Karen le pide su nombre de usuario de Facebook a José y éste contesta José López, ella utiliza su iPhone para buscarlo en ésta red social, pero se encuentra con la sorpresa que hay 200 usuarios con el mismo nombre, por lo que tiene que hacer un repaso por las fotos de perfil hasta que logra encontrarlo y envía la solicitud de amistad pero debe esperar a que José acepte.

Target:

- Hombres y mujeres
- 18 a 45 años de edad,
- Clase media baja, media alta y alta.
- Estudiantes y profesionistas
- Gusto por la tecnología.

Requerimientos:

1. Aplicación que sea capaz de leer códigos QR con la cámara frontal de preferencia.
2. La aplicación debe guardar tarjetas de presentación que serán representadas en una imagen QR. Dichas imágenes se pueden guardar (compartir) como imagen o ser usadas en la aplicación para que sean leídas por otro usuario mediante la misma.
3. Dos usuarios con la misma aplicación podrán leer simultáneamente sus códigos QR y con ello se agregan como contactos, uno del otro.
4. Se requiere que el método de registro sea el más sencillo: redes sociales o email y contraseña.
5. La cuenta del usuario debe ser multi-dispositivo, es decir, el usuario podrá ver sus contactos, tarjetas y perfil en cualquier dispositivo en que se registre.

6. Si un usuario lee el QR de una tarjeta (en un slide de presentación, papel, pantalla, con la aplicación) entonces se le preguntará si quiere compartir una tarjeta con el dueño del QR. El dueño del QR podrá ver esas tarjetas en una pantalla para conocer quien, donde y en qué fechas leyó su QR.

7. El usuario se podrá conectar con Facebook, Twitter o en su defecto se registrará con email y contraseña. Si el usuario se registra con Twitter deberá proporcionar un email. Los usuarios que se conecten con twitter o email y contraseña por primera vez serán registrados automáticamente y se les mandará un correo para que activen la cuenta.

8. La primera vez que se abra la aplicación (o si no existe sesión abierta) entonces se solicitará al usuario que se registre, si el usuario es nuevo usuario pluzer entonces la app le solicitará llenar sets de datos (teléfonos, emails, domicilios, redes sociales) de su perfil así como obligatoriamente crear una tarjeta.

9. Cada dato del perfil tendrá una etiqueta para identificarlo (por ejemplo: teléfono casa, teléfono oficina, teléfono personal, etc.), se deberá proporcionar un set de etiquetas (por ejemplo: casa, oficina, personal, telcel personal, movistar trabajo, etc.) predefinidas para elegir una o bien se puede crear una personalizada.

10. Los únicos campos obligatorios en todas las tarjetas son nombre y apellido (campos default).

11. El usuario podrá tener un número ilimitado de tarjetas de presentación, todas ellas contarán con los campos default y el usuario podrá agregar más campos al momento de crearla o editarla. Las tarjetas tendrán dos representaciones: QR Pluzer (para hacer pluzer con un usuario) y QR Pluzer externo, éste último se podrá guardar como imagen para ser leído por la misma app pero de forma externa, sin necesidad de hacer pluzer.

Para crear las tarjetas de una forma sencilla y rápida el usuario contará con un set de datos (teléfonos, emails, direcciones, redes sociales) que le servirá para ir creando tarjetas personalizadas para distintas ocasiones: trabajo, familia, amigos, desconocidos, conferencias, etc.

12. Al momento de que un usuario lea un QR Pluzer externo se le preguntará si quiere compartir una tarjeta personal con usuario dueño del QR Pluzer externo.

13. El usuario podrá ver, editar y eliminar sus tarjetas de presentación así como exportarlas-compartirlas como imagen.

14. El usuario podrá agregar a un contacto mediante el método pluzer o el método unidireccional.

Pluzer ó Bidireccional: colocando dos smartphones, uno enfrente del otro. Cada smar-

phone leerá el código QR del otro dispositivo con la cámara frontal. O en su defecto si alguno no cuenta con cámara frontal entonces uno teléfono leerá un QR y el otro después leerá el correspondiente.

Unidireccional: Un usuario exporta su tarjeta en forma de QR y lo imprime y otro usuario lee el QR para agregarlo como contacto.

15. El usuario podrá ver, editar y eliminar los contactos.

16. El usuario podrá usar los datos de sus contactos para: llamar al teléfono, agregar el teléfono a la agenda del smartphone, mandar email, agregar como amigo a facebook, seguir en twitter, ver domicilio en mapa.

17. El usuario podrá ver las tarjetas de los usuarios que han leído sus QR externos.

18. En la configuración el usuario deberá elegir el comportamiento de la app al momento de agregar un contacto. Si desea que automáticamente se siga al contacto a las redes sociales (previamente asociadas a la app) y si se desea que el contacto sea agregado a la agenda del teléfono. Si el usuario no desea que sea automáticamente entonces podrá hacerlo manualmente.

19. Para que un usuario pueda crear, editar una tarjeta o actualizar su perfil debe contar con conexión a Internet.

20. El usuario podrá cerrar sesión en un dispositivo y podrá cancelar su cuenta de forma definitiva.

21. Cuando el usuario borre una tarjeta o cancele su cuenta habrá un efecto en cadena sobre la(s) tarjeta(s) involucrada(s) de dicho usuario que se encuentre(n) como contacto en otra cuenta.

22. Para poder tener un perfil activado y que el usuario pueda pasar a la pantalla de tarjetas (la primera vez que entra a la app registrado) que se requiera guardar un campo adicional a nombre y apellidos.

23. Para poder pasar a la pantalla de pluzer_no_cam el usuario debe tener creada al menos una tarjeta.

Personal requerido: Administrador de proyecto, desarrollador Android, desarrollador iOS y diseñador gráfico.

4.3 ARQUITECTURA DE LA INFORMACIÓN

Es el esquema de organización y navegación por los contenidos resultantes de los requerimientos para que el usuario encuentre y gestione de manera efectiva la información, puesto que en la fase de análisis únicamente se trata de identificarlos más no se hace una clasificación o jerarquización de ellos. Asimismo especifica las relaciones que habrá entre los distintos contenidos, así como las posibles interacciones entre ellos.

Las tareas que desempeña el arquitecto de la información deben asegurar el correcto diseño de:

- El sistema de clasificación de la información.
- La estructura de navegación.
- El etiquetado o rotulación de las opciones.
- El sistema de interacción de contenidos.

Primeramente el arquitecto de la información esta encargado de analizar los contenidos, para distinguir cuales pueden ser agrupados o divididos y de ser así hasta que grado pueden ser descompuestos sin perder el sentido; los componentes que surjan podrán ser clasificados de acuerdo a la importancia de su información. El volumen de contenidos que deben incluirse en la aplicación móvil afecta directamente la complejidad del trabajo del arquitecto.

ESTRUCTURA DE NAVEGACIÓN

La estructura de navegación es la disposición y orden de las partes que componen la app, conforme a Monjo (2011) los tres tipos básicos de estructuras son: jerárquica, lineal y red, aunque pueden establecerse estructuras combinadas (p.168).

Estructura jerárquica. Es la que a partir de una pantalla o menú inicial se accede ordenadamente a los diferentes niveles y subniveles de contenido, sus principales características son:

- Se trata de una estructura muy recomendable para grandes volúmenes de contenido.
- Permite al usuario orientarse bien en la navegación, sabiendo en qué nivel de contenido se encuentra.
- Permite ordenar los contenidos en varios niveles de profundidad: lo más general en niveles superiores, más específico en los más profundos. Lo mejor es no superar tres niveles de profundidad a partir de la pantalla inicial; si no es posible, los menús más

amplios tienen que corresponder al primer nivel para que el usuario localice la mayor parte de contenidos en la página principal.

Estructura de navegación lineal. Es muy simple al únicamente proporcionar de manera secuencial la información. Tiene las siguientes características:

- El contenido progresa de lo general a lo específico.
- Se trata de una navegación altamente pautada.

En este tipo de navegación, la mejor experiencia por parte del usuario se obtiene cuando se combinan tres tipos de enlaces:

- 1) Enlaces de vecindad, que permiten desplazarse horizontalmente, es decir, entre categorías del mismo nivel.
- 2) Enlaces de retorno, que permiten volver inmediatamente a determinadas páginas anteriores, como el home o menú principal.
- 3) Enlaces de índice, que permiten saltar directamente a páginas de información específica.

Estructuras de navegación en red. Aparentemente sin un orden con las siguientes características generales:

- No existe jerarquía.
- Navegación muy libre.
- Requiere una alta cantidad de enlaces.
- Puede provocar desorientación en el usuario.

Ya puntualizada la estructura de la aplicación, se representa visualmente mediante diagramas; estos deben especificar a detalle la macroestructura de la app con base en los requerimientos definidos anteriormente, para que todos los miembros del equipo tengan una visión general y puedan dar su punto de vista. El detalle con que se hacen estos diagramas es responsabilidad del diseñador y su prioridad es indicar cómo navega el usuario a través de tareas determinadas y qué pasos conllevan a realizar tal tarea. No obstante los detalles gráficos correspondientes a la interfaz todavía no serán representados.



Todo el contenido de la app fue estructurada en 5 principales acciones, cada una representada en una pantalla que a su vez está dividida en otras más que permiten la edición u obtener más información de los contenidos.

Contactos: ésta función almacena los datos personales de los usuarios con los que se ha hecho “pluzer; contiene información como nombre, dirección, mail, teléfono, trabajo, Facebook, Twitter, fecha y lugar donde se conocieron, etc. ”

Invitar amigos: para que una aplicación logré difundirse rápidamente el usuario satisfecho debe recomendarla en los mayores medios posibles, en este caso redes sociales y así poco a poco ir atrayendo más personas, por ello es importante que ésta acción de invitar este muy a la mano del usuario, el objetivo es lograr que éste sea el nuevo concepto de tarjeta de presentación.

Pluzear: La principal meta de la app consiste en leer códigos QR que contienen la información del usuario, a través de la cámara del dispositivo; con 3 sencillos pasos se logra el objetivo: primero apretar el botón correspondiente para que se active la cámara, después seleccionar la tarjeta que se desea compartir y por último esperar a que el teléfono reconozca el código QR y listo.

Tarjetas: con dicha acción el usuario crea y personaliza las tarjetas que compartirá, para ello puede seleccionar los datos que contendrá (nombre, domicilio, teléfono, trabajo, etc.) y nombrarla de acuerdo al tipo de contenido o a criterio personal. Puede agregar, eliminar o editar cuantas tarjetas desee.

Configuración: esta función se refiere a los ajustes de la aplicación para lograr su funcionamiento adecuado de acuerdo a los criterios del usuario, éstos están relacionados sobre todo al comportamiento de las redes sociales y al cierre de sesión.

Los diagramas están compuestos por wireframes (cuadros de alambre) que representan cada pantalla individual de la aplicación móvil de manera simplificada, lo que posibilita tener una idea inicial de la organización de los elementos que contendrá, estableciendo informativos e interactivos. Es de mi agrado la comparación que realiza Cuello (2013) al mencionar que “el wireframe es para la app lo que el plano arquitectónico para una casa; en este plano se pueden visualizar los espacios y elementos funcionales de una forma clara y simplificada”.

Siguiendo con esta semejanza de planos, los wireframes están dibujados de forma lineal y de un solo color, permitiendo centrarse en la estructura o esqueleto de una pantalla y por el momento dejar a un lado la estética de la app. Irene Pereyra Directora Global de UX y Estrategia en Fi, respondiendo a una entrevista para el libro Diseñando apps para móviles recomienda: no pasar directamente al diseño posibilita enfocarte en cosas como la funcionalidad, la interacción de la estructura general y el viaje que quieres que el usuario haga cuando navega a través de tu experiencia interactiva. Omitir esta recomendación es



ESTRUCTURA DE NAVEGACIÓN

extremadamente peligroso porque te puede llevar a concentrarte solo en los detalles de diseño (texturas, sombras, volúmenes, etc.) que, aunque son importantes, deberían considerarse en una etapa posterior del proceso, una vez que la estructura de interacción ha sido determinada.

INTERACCIÓN

Es importante contemplarla durante esta etapa de wireframes, porque aún no estás comprometido con ningún tratamiento visual y es mucho más fácil moverse rápidamente a través de diferentes conceptos e ideas sin tener que modificar cada elemento por cada estructura.

El sistema de interacción de contenidos se refiere a la comunicación que existe entre los distintos elementos que componen la interfaz; cada sistema operativo tiene sus propios parámetros de interactividad lo que le permite tener su propia identidad y hace que la experiencia de usuario sea diferente a las demás. Por ejemplo la barra de navegación en Android suele estar en la parte de arriba para evitar el contacto con los botones físicos de atrás y menú de los smartphones, mientras que en iOS esta barra se coloca en la parte de abajo por no tener problema con botones físicos.

Además de servir como canal de comunicación para navegar por las diferentes secciones de la app, los elementos de navegación permiten al usuario ubicarse dentro de ésta y saber hacia donde va en cada momento; la orientación es fundamental para el funcionamiento óptimo.

Según Hassan (2002) consultor y diseñador de interacción, para que el usuario no se desoriente al navegar por el sistema, resulta fundamental respetar dos principios:

- 1) El usuario debe encontrar fácilmente lo que busca, ya que de lo contrario se aburrirá navegando y abandonará la aplicación.
- 2) El usuario ha de saber en qué zona de la aplicación se encuentra, o si realmente sigue estando en nuestra aplicación (se da sobre todo cuando la app requiere utilizar el navegador web para mostrar algún contenido). Los factores que conforme al mismo autor permiten que el usuario se mantenga orientado son los siguientes:

Simplicidad visual. Conlleva ser mínimo en el número de elementos, donde lo más importante es que aquellos que aparezcan en la interfaz tengan una función claramente definida, colaboren a cumplir el objetivo de la app y sobre todo ayuden al usuario.

Los dispositivos móviles no se prestan para mostrar mucha información en pantalla, por tal motivo, la simplicidad también consiste en el aprovechamiento del espacio al manejar la economía visual y poseer un buen criterio para determinar qué incluir en el diseño. El

contar con una gran cantidad de elementos en pantalla llega a agobiar al usuario, es preferible mostrar solo lo necesario en ese momento y en esa situación de uso. Hacer el diseño simple no es tarea fácil, pero reporta grandes beneficios en la experiencia de uso de la aplicación, este factor está ligado completamente con la usabilidad.

Consistencia visual. La consistencia en el diseño es fundamental para que el usuario pueda reconocer en todo momento si sigue en nuestra app, es el sello que distingue a todos los contenidos que forman parte de ella. Esta consistencia se refiere tanto a la estructura de las pantallas, como a la gama cromática, la tipografía utilizada, los íconos, etc.; en pocas palabras es la identidad de la aplicación.

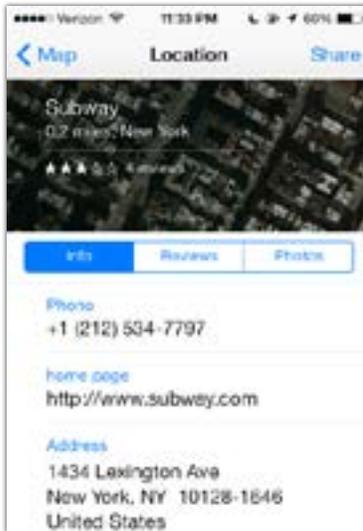
Navegación intuitiva. La forma de navegar entre contenidos, debe resultar fácil de comprender para el usuario, evitando la sensación de desorientación que puede ocasionar una navegación confusa. Este factor está vinculado con la consistencia visual por lo mencionado anteriormente en sistemas de interacción, de que cada sistema operativo propone diferentes elementos para navegar por la app como barras, botones, pestañas; al hacer uso de ellos el usuario ya estará familiarizado a primera vista y sabrá como ir de una sección a otra.

Aunque la identidad del sistema aporte un porcentaje alto de intuición, para el usuario es significativo saber y prever qué pasará al pulsar un botón o que cambiará en la pantalla. El saber donde se está dentro de los contenidos de la app y como volver hacia atrás son características muy importantes que ahorran tiempo y esfuerzo en comprender cómo ir de un sitio a otro, además de permitir un uso fluido de la aplicación.

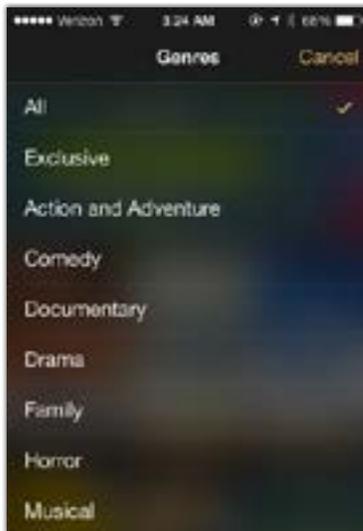
Jerarquía visual. Es el orden en que se interpretan los elementos que conforman la pantalla; en occidente lo habitual es leer de izquierda a derecha y de arriba hacia abajo. Por tal motivo los usuarios interpretan a los elementos que se encuentran más cerca de la esquina superior izquierda de la pantalla con mayor nivel jerárquico, y el resto de elementos que se encuentran en otra posición suelen entenderse como subniveles de los primeros. La jerarquía visual no solo se logra con la estructura, puede reforzarse como se dijo en el capítulo anterior con recursos como el tamaño, el color o la tipografía.

Breadcrumbs (migas de pan). Se trata de un recurso muy utilizado en sistemas con contenidos en los que existen muchos niveles, ya que permite que el usuario sepa dónde se encuentra, y de dónde viene (cuáles son los niveles previos a su estado). En iOS sobre todo es donde se utiliza este recurso al no tener botón físico de atrás, el botón asignado para ello en lugar de tener la leyenda “atrás” suele tener la leyenda de la sección anterior.

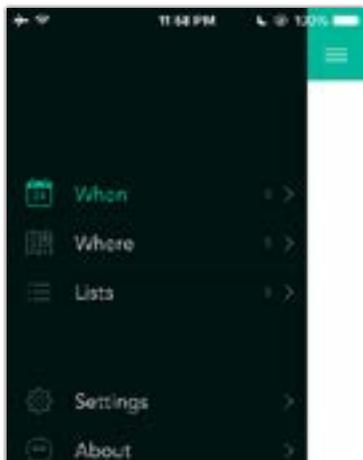
Redundancia. Es importante que no existan dos opciones en una misma pantalla que activen la misma función; el usuario interpretará que existe alguna diferencia entre ellas, y se desorientará al ver que llevan al mismo sitio. O caso contrario si dos botones son muy parecidos, el usuario no sabrá que función activará.



PESTAÑAS



LISTAS



MENÚ TIPO CAJÓN

Cerrando este tema, es conveniente conocer los patrones de interacción para utilizar el que mejor se adapte a las necesidades de la app, se les llama patrones porque son una solución de diseño que se ha demostrado que funciona más de una vez, Prieto (2011), hace una muy buena descripción de ellos (p. 175):

PESTAÑAS

Se utilizan para elegir secciones o cambiar entre pantallas que, de acuerdo a la estructura de la aplicación, tienen el mismo nivel de jerarquía, mostrando siempre en dónde se está y hacia dónde más se puede ir. En este y todos los patrones es necesario destacar la pestaña o botón seleccionado, sin alterar el orden y la ubicación inicial de estos elementos, además de únicamente servir como elemento interactivo.

LISTAS

Es una forma de mostrar resumidamente tantos ítems como sea necesario dispuestos verticalmente, que permite al usuario obtener mayor información al tocar alguno de ellos.

En las listas se pueden mostrar tanto textos como imágenes, siempre ordenándolos de acuerdo a su contenido, jerarquía o como muchas veces se hace alfabéticamente; el ejemplo más común son las listas de contactos telefónicos ordenados de la A a la Z y que al seleccionar cierto nombre despliega las opciones de llamada o mensaje.

MENÚ TIPO CAJÓN

Este patrón ha sido popularizado en los últimos meses sobre todo por las apps de redes sociales, permite cambiar rápidamente entre pantallas de la aplicación; al pulsar un botón o deslizando el dedo de izquierda a derecha de la pantalla, se despliega de manera lateral una lista con las secciones más importantes, oculta hasta ese momento.

Las ventajas de este tipo de menú son el mejor aprovechamiento del espacio y cuando es desplegada la lista, brinda una forma cómoda para elegir contenidos. La única posible desventaja es que quien no este familiarizado con este patrón, le será difícil encontrar este menú y por lo tanto solo sabrá cuales son las opciones disponibles.

BARRA DE ACCIONES

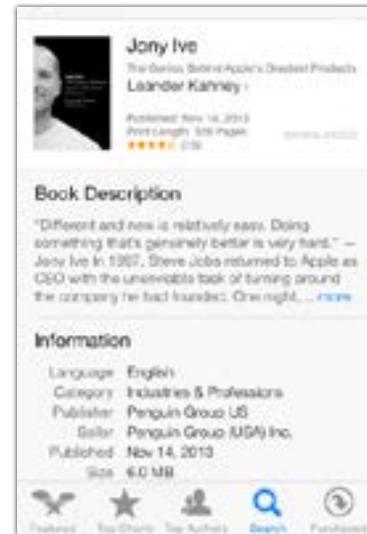
En todos los sistemas, el compendio de acciones que se pueden realizar se representa por medio de íconos, por ello la correcta selección de estos recursos gráficos es fundamental. Hay diferentes sitios donde ubicar las acciones de acuerdo a su jerarquía y funcionamiento, donde las más importantes están visibles y las menos, ocultas. Donde sea que se encuentren, la posición tiene que ser consistente a través de las distintas pantallas y con otras aplicaciones del sistema operativo. Este patrón suele ser sustituido por el menú tipo cajón aunque existen casos donde se ocupan ambos como en la app de Facebook.

BUSCAR

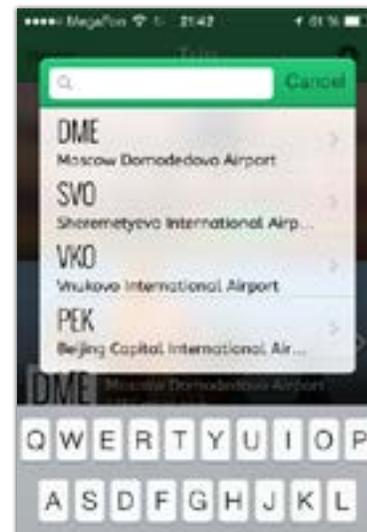
Teniendo en cuenta que uno de los usos principales del móvil es el consumo de contenidos, la herramienta buscar es una manera esencial para llegar a ellos. En apps que muestran grandes cantidades de datos, la búsqueda puede ser incluso la función primaria. Cuando es utilizado en listas evitan la tediosa tarea de ir desplazando hacia abajo la pantalla un sin fin de veces.

CUADROS DE DIÁLOGO (POP OVER)

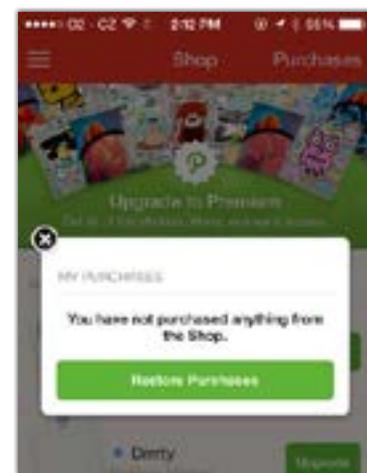
Hay casos puntuales en los que hay que interrumpir al usuario de forma temporal para que tome una decisión o para explicarle mejor algo que ha sucedido antes de continuar una tarea. Mientras los diálogos están visibles en pantalla no es posible hacer otra cosa en el resto de la aplicación. En este sentido es primordial ser claro con las opciones a elegir, pues el usuario debe estar convencido de su elección y lo que sucederá a continuación.



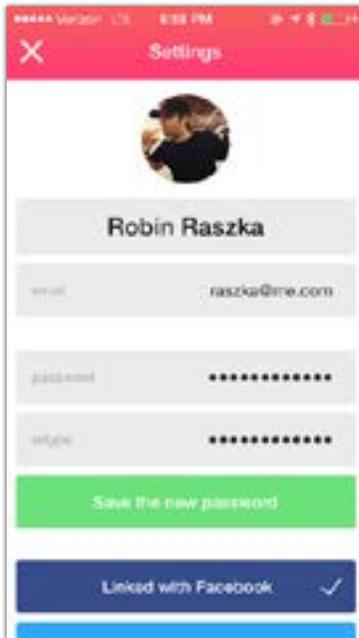
BARRA DE ACCIONES



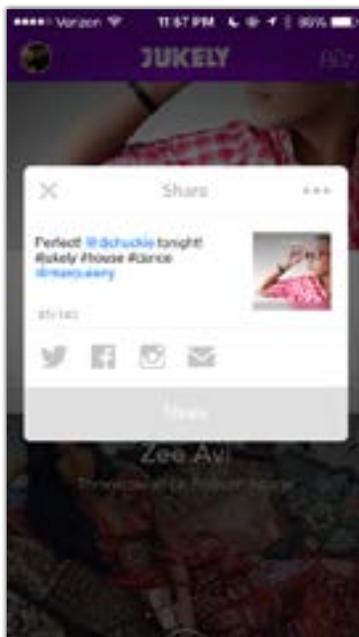
BUSCAR



POP OVER



DATOS



COMPARTIR

Cuando se trata de cuadros que contienen avisos que no requieren una toma de decisión, estos son informativos y tienen únicamente un botón que se ocupa de cerrarlos. Es recomendable limitar su uso para mensajes graves o trascendentales que no puedan esperar.

INTRODUCCIÓN DE DATOS

La introducción de datos en el móvil puede ser tediosa cuando se trata de campos que requieren el uso del teclado, un elemento que ocupa gran parte de la pantalla y que dificulta la navegación entre los campos para introducir información; en este caso lo mejor es prevenir que tanto afectará la interfaz, haciendo pruebas sobre los posibles elementos que desaparecerían o cambiarían de tamaño.

COMPARTIR

Probablemente sea una de las acciones más empleadas en estos días: compartir contenidos con amigos, en Facebook, en Twitter, por mensaje de texto, como sea. Los sistemas operativos también han notado esta necesidad y han facilitado una implementación integrada al sistema muy fácil de aprovechar; es un botón que al presionarlo despliega las apps en las que se puede visualizar dicho contenido.

4.4 DISEÑO GRÁFICO

En ésta fase se definen y realizan las características gráficas de la interfaz, tomando en cuenta la información reunida en las fases de análisis y arquitectura tomando en cuenta los wireframes realizados con el tipo de interacción que se desea para la app.

Siguiendo el método de desarrollo combinado con los métodos de diseño mencionados en el capítulo 2, los pasos que se requieren para lograr el diseño de la interfaz suelen ser los siguientes:

1) ANÁLISIS DEL LIBRO DE ESTILO O DE LOS DETERMINANTES GRÁFICOS APORTADOS POR EL CLIENTE.

El cliente suele tener una idea definida de la imagen que desea mostrar; en caso de que se trate de una organización o empresa, pueden existir requisitos de imagen que deben respetarse, generalmente plasmado en el manual de identidad. Es muy importante consultar si existe el libro de estilo, ya que de ser así se tiene que respetar y podría no existir libertad para diseñar.



Ésta app no cuenta con una identidad visual por lo que el diseño se empezará desde cero; las recomendaciones que se dieron por parte del cliente es que tenga estilo minimalista dirigida hacia la nueva tendencia del diseño plano (flat), además se busca que el usuario perciba una interfaz muy sencilla y con tan sólo unos clics logre su objetivo.

2) DOCUMENTACIÓN GRÁFICA

Antes de empezar a definir las cuestiones relacionadas con el diseño gráfico, se lleva a cabo un análisis del tipo de contenido que integrarán la aplicación (texto, fotografías, esquemas, gráficas, ilustraciones, etc.) debido a que cada tipo podría tener distinta forma de visualización, ocupar más espacio de la pantalla o no ir acorde al resto de la app.

Por otro lado, el diseñador se documenta acerca del tratamiento previo de contenidos similares, buscando ejemplos de aplicaciones que previamente hayan abordado el mismo tema o que se hayan dirigido al mismo tipo de usuario; la observación de producciones existentes es importante porque permite recopilar ideas y tener en cuenta aspectos que se hubieran podido pasar por alto, además de servir como inspiración para nuestra app.



Las apps que sirvieron como guía fueron Cobook, About.me y Likify.



ABOUT.ME



COBOOK



RETÍCULA DE LA APP PLUZER

las guías de interfaz de usuario de los diferentes sistemas operativos para ser el tamaño mínimo recomendable; y para determinar el espacio entre cada elemento de la interfaz de usuario este debe surgir a partir de múltiplos de 8 px.

4) ELECCIÓN DE LA GAMA CROMÁTICA

En el capítulo 3 ya se han mencionado los aspectos más importantes a considerar tanto técnicos como visuales para la toma de decisión del color. Dando un pequeño resumen, a partir de los determinantes de estilo o del tratamiento que se quiere dar a los contenidos, se define el tono cromático general de la aplicación según el tema, el tono general que se quiere transmitir y el tipo de usuario. Si existe un libro de estilo o una imagen de marca, normalmente la aplicación deberá respetar la gama de colores predeterminada.

Por parte del cliente se pidió un color claro como fondo, el color elegido es el blanco porque da la sensación de limpieza en el diseño y genera mayores espacios. Para dar un ligero contraste con el fondo se utilizó una escala de grises para resaltar algunos ele-

3) DISEÑO DE RETÍCULA

Formando parte de la definición de los elementos principales del diseño (color y tipografía), es fundamental definir desde el principio la retícula en la que se basará la aplicación, probablemente desde los wireframes ya exista un esbozo de ella. El establecimiento de una retícula resulta más importante porque asegura mayor equilibrio en nuestra composición al momento de diseñar; es una estructura en dos dimensiones hecha de líneas verticales y horizontales utilizada para estructurar el contenido, sirve como una armazón para organizar textos, imágenes o videos de una forma racional y fácil de visualizar. El diseño de la retícula está en relación directa con el volumen de los contenidos y con el tipo de navegación definida en la arquitectura.

En promedio, 48 px es la medida del módulo del que se parte para crear los elementos en pantalla, porque este tamaño garantiza que el usuario sea capaz de presionar con sus dedos cualquier botón de forma fiable, cómoda y precisa. Esta medida ésta respaldada por

mentos que refuerzan el texto y también elementos de interacción que muestran los campos seleccionados.

El color rojo proporciona el contraste en todas las pantallas de la interfaz, tanto en la barra superior como en los elementos más importantes activos; elegí este color por ser vibrante y calido, es fácil de recordar e identificar además de representar esa calidez que se da cuando dos personas se conocen. También es parte del ser diferente de las demás redes sociales actuales donde predomina el color azul.



GAMA DE COLORES PARA PLUZER

5) ELECCIÓN DE LA TIPOGRAFÍA

De igual forma que el color, el manejo de la tipografía ya ha sido descrita en el capítulo anterior. En líneas generales, se aconseja utilizar un máximo de dos tipos de letra distintos, ya que la mezcla de muchas variaciones puede implicar poca legibilidad y un aspecto visual caótico; si se utilizan dos tipos de letra distintos, es importante que sean lo suficientemente distintos como para percibirse como tales, las combinaciones que ofrecen mayor contraste son las de un tipo de palo seco, con un tipo romano. El contraste entre fondo y tipografía es el más importante a cuidar.



La tipografía utilizada es Helvetica Neue en sus variantes Light y Regular, esta fuente cuenta con muy buena legibilidad al ser palo seco, no cuenta con adornos y va de acuerdo a la sencillez de la app; su éxito está comprobado al ser la preferida de la plataforma iOS y no causa conflicto en Android, al semejarse mucho a la fuente Roboto.

6) GENERACIÓN DE LOS PRINCIPALES ELEMENTOS DEL DISEÑO.

A partir de la división y asignación de espacios de pantalla, se concretan los principales elementos, que incluyen:

a) El tratamiento de los fondos.



 Anteriormente mencione que el fondo es en color blanco totalmente plano, por lo que no existió mucha labor en este punto.

b) La definición de los principales bloques de texto (títulos, subtítulos, índices y menús). Se define el tipo, color y tamaño de letra de cada categoría de texto.

 Barra superior: Helvética Neue Regular 36 pts.
Títulos: Helvética Neue Regular 36 pts.

Categorías: Helvética Neue Light 24 pts.

Menús pops: Títulos Helvética Neue Light 30 pts.

Categorías Helvética Neue Light 30 pts.



c) La integración de logotipos.

 El logotipo de la aplicación no sólo está presente en el ícono del launcher, sirve como botón para realizar la acción más importante “pluzear”, por lo que fue de los primeros elementos a elaborar.

En las primeras propuestas se jugó un poco con el símbolo “+” y los pequeños cuadros representativos de los QR, simulando la conexión que genera entre personas en esta app, sin embargo remontaban demasiado a la cruz religiosa. Después se realizaron pruebas con síntesis de personas formando un círculo con el “+” en el centro, al principio era agradable pero daba la sensación que no era una forma muy fácil de reconocer.



El logotipo final surge de la idea de que por lo regular las redes sociales utilizan su letra inicial para identificarse, tomando esto de referencia y agregando el símbolo “+” como elemento de apoyo se dio el resultado; en los primeros esbozos aún no estaba clara la idea pues no sabía como combinarlos, el “+” formaba parte del ojal, posteriormente surgió la idea de poder encajar perfectamente como parte del asta de la “P” y fue así como quedé convencido. Así mismo está acompañada por el nombre completo de la aplicación para aparecer dentro



PROPUESTAS LOGO

del launch screen, en este caso la tipografía tiene una separación en el centro para darle el mismo estilo que la letra “P” al estar formada por dos figuras.

d) Barras de acciones.



Son cinco los elementos utilizados en esta barra, cada uno representado por un ícono diferente, recordando que la acción “pluzear” debe destacar de acuerdo a los requisitos pedidos por el cliente. Por tal motivo cuando un botón está desactivado aparece únicamente el contorno del ícono, cuando es activado el ícono cambia de color y tiene relleno.

La primera acción “Contactos” está interpretado por el ícono de una persona, este recurso es utilizado por la mayoría de apps, sobre todo en agendas telefónicas; y si se uso de esta manera no es por falta de creatividad, es porque es recomendable emplear elementos que el usuario ya identifica.

Como segunda acción esta “Invitar amigos”, en ella se empleo el mismo ícono de la persona pero esta vez acompañada de otra para asimilar a una pareja de amigos, agregando el símbolo “+” para dar el sentido de agregar, incorporar más usuarios.

La tercera acción corresponde a “Pluzer”, es la principal acción de la app y como se menciona previamente ésta conformada por el logotipo, resalta de las demás por su color y tamaño.

En cuarta acción está “Tarjetas”, para lograr el ícono lo que se realizó fue abstraer la pantalla de ésta función, que da el aspecto de una tarjeta real; lógicamente al incluir texto iba a ser muy pequeño y no alcanzaría a distinguirse por ello lo mejor fue representarlo como líneas.

“Configuración” es la última de las acciones de la barra, su ícono corresponde a un engrane, dando uso de las metáforas como principal recomendación al momento de diseñar íconos; éstas se refieren a describir o calificar algo a través de su semejanza o analogía con otra cosa, en este caso un engrane se utiliza para ajustar o hacer funcionar algo, tarea que precisamente realizará esta acción, adecuar las opciones a como el usuario desea que actúe la app.

La barra que contiene estas acciones fue algo difícil de lograr, porque en los primeros diseños perdía protagonismo el botón de pluzer; el que éste botón fuera más grande que los demás no conseguía tener mayor jerarquía y menos cuando una acción estuviera activada. En otro intento el botón principal fue de color diferente al resto de acciones pero generaba confusión entre cual era la función requerida. En la propuesta final se consiguió el objetivo al poner el botón pluzer con una envolvente circular del mismo color de fondo y con tamaño mayor respecto a los otros botones para darle mayor peso, pero si usando el mismo color gris de contorno que los demás y usando el color rojo en todos los botones cuando cualquiera de éstos esta activado.



PROPUESTAS PARA BARRA DE ACCIONES

7) GENERACIÓN DE LOS ELEMENTOS SECUNDARIOS

Una vez concretados los elementos que caracterizarán la aplicación, se pasa a detallar el diseño de los elementos secundarios, tales como:

a) Botones. El tratamiento de las opciones de pantalla debe ser coherente con el estilo visual de la aplicación. Los botones pueden ser representados de dos maneras distintas, por medio de íconos o de texto sobre alguna envolvente.

 La mayoría de los botones son pequeños menús con dos opciones, en los que solo se puede elegir una de ellas; estos elementos simulan los interruptores eléctricos en las que está una u otra función activada dependiendo del lado en que se haga presión. Su representación es por medio de texto para cada opción encerrado en un rectángulo, que al juntarlo forman el interruptor.



DETALLE DE LOS BOTONES USADOS EN PLUZER

El resto de los botones, representan funciones como llamar, eliminar y nuevo, y son interpretados por metáforas muy conocidas por los usuarios, un teléfono para llamar, un bote de basura para eliminar y un símbolo de más para algo nuevo.

b) Texto de contenido. Se define el tipo, tamaño y color de letra del texto correspondiente al contenido de la aplicación.

 Información: Helvética Neue Light 30 pts. Color Negro #000000
Intrucciones: Helvética Neue Light 30 pts. Color Gris #f2f2f2

c) Imágenes de contenido. Se determinan los formatos y las diferentes aplicaciones de las imágenes de contenido. En aplicaciones de gran volumen, es importante determinar un número máximo de variaciones de formato y localización, para facilitar el trabajo del equipo de edición.



Son únicamente utilizadas en las fotos de cada contacto, dependiendo de la pantalla en la que aparecen es su tamaño y siempre están dentro de una mascarilla circular.

d) Icono de la aplicación. Es diseñado al último porque debe estar acorde al estilo de la app, además de resumir en una imagen el contenido de la aplicación.

Todas las características de diseño de una app tienen un denominador común, la sencillez. Limitar al mínimo colores, tipografías y estampados para conseguir un resultado eficaz, que represente a la perfección la idea de la aplicación.

Un buen icono de aplicación es primordial para causar buena impresión y animar a los usuarios a descargársela, por eso debe ser llamativo.



Correspondiendo a lo sugerido en el capítulo 3, el icono únicamente estará conformado por el logo de Pluzer en tamaño grande para ser perfectamente legible y lógicamente aparecera con los colores representativos de la app.



ICONO PARA TIENDAS Y MÓVILES

8) PROBAR EN EL MÓVIL

Si bien es cierto que este paso no es mencionado en el método de desarrollo, es un buen consejo por parte de Cuello (2013), el menciona que una interfaz puede verse muy bien en la pantalla del ordenador, pero este no es un escenario real pues no es allí donde finalmente se verá el diseño. Es apropiado, a medida que se va diseñando, probar constantemente en un smartphone acorde al sistema operativo en que se desarrollará, para verificar la visualización y no esperar a tener implementada la aplicación después del código, para tener que corregir todo.

Implementar este paso y acostumbrarse a hacerlo de forma habitual en este proceso de diseño, posibilita estar seguro de que la tipografía, el tamaño de los elementos y el color son correctos y que los contrastes están a buen nivel. Una herramienta muy útil para llevarlo a cabo es la aplicación Dropbox en su versión móvil, debido a que permite visualizar los mockups en pantalla completa del smartphone, creando una muy buena simulación de la app, además de que ante cualquier cambio solo se tiene que sincronizar la nueva imagen. Cabe aclarar un mockup es el modelo a escala o tamaño real de un diseño, utilizado para la demostración, evaluación del diseño únicamente de manera visual,

en este caso es la representación de cada pantalla y a diferencia de los wireframes si cuenta con un diseño detallado de sus elementos.

9) PREPARAR LOS ARCHIVOS

Ya definido el diseño de la interfaz y autorizado por el cliente, la siguiente etapa consiste en separar y preparar los archivos listos para que el desarrollador pueda implementarlos, este paso también es denominado separación de piezas. Habrá casos en que no todos los elementos gráficos sean piezas, pues el desarrollador puede utilizar parte de los elementos predefinidos en el programa en que desarrolle que cumplan con los requisitos de la interfaz. Hacer esto de la forma correcta favorece una implementación fiel del diseño.

La variación en las características del hardware móvil sobre todo en las pantallas, las cuales ya fueron citadas en el apartado de resoluciones de pantalla del capítulo anterior, afectan el trabajo del diseñador en este paso, en el aspecto en que debe crear piezas para cada clasificación de tamaño en Android o para cada dispositivo en iOS.

Como se trabaja con muchas imágenes diferentes, por cada resolución, es importante mantener la organización y consistencia en la forma de nombrar los archivos exportados. Trabajar de esta manera permite buscar imágenes de manera rápida y con solo el nombre debería de ser posible identificar la función que cumple o la pantalla en que será utilizada.

En Android se recomienda usar carpetas diferentes para cada una de las densidades; allí se almacenarán las imágenes manteniendo el mismo nombre para cada una de los casos. Como ayuda adicional a las carpetas, las imágenes pueden ser nombradas con tipo de pieza y cual es su función. Por ejemplo, los íconos pueden empezar con iconoamigos; incluso, pueden sumarse la sección a la que pertenecen para dar más claridad acerca del tipo de imagen. Un gráfico que se llame iconoamigosmenu.png nos dará un indicio bastante preciso de qué es lo que hace y dónde se ubica.

Las imágenes que se exportan para iOS también tienen una forma especial de nombrarse dependiendo de si están destinadas para terminales con pantalla retina o no retina. En el primer caso hay que incluir @2x después del nombre. Por ejemplo, tendríamos la imagen botonpausa@2x.png para retina y botonpausa.png para no retina; y como en el caso de Android lo mejor es crear carpetas para cada dispositivo (iPod, iPhone, iPad, etc.).

4.5 DESARROLLO

La siguiente fase le corresponde al desarrollador, en ella se encarga de dar vida a los diseños a través de la programación de código dentro del entorno de desarrollo correspondiente a la plataforma del sistema operativo (Xcode, Eclipse, etc.); crea la estructura sobre la que se apoya el funcionamiento de la app.

Además de las habilidades técnicas que posea el desarrollador, es primordial tener buena comunicación constante y fluida con el diseñador, para asegurar una correcta implementación. Es muy común que aparezcan dudas sobre la retícula, el tamaño de gráficos, fuentes, el color exacto a utilizar, piezas faltantes o cosas que van resultando evidentes hasta que se programan; y a pesar de tener como guía los mocks, es mejor que el propio diseñador aclare esta clase de dudas.

Saltando un poco las siguientes fases del proceso para cerrar este tema del desarrollo, una vez terminada la versión inicial, prototipo de alta fidelidad, y tras los resultados arrojados por la evaluación de usabilidad y funcionamiento, el desarrollador se dedicará a corregir errores funcionales (bugs), para garantizar el desempeño de la aplicación móvil y permitir su aprobación en las tiendas.

4.6 PROTOTIPO

Es una fase no necesariamente después del trabajo del desarrollador, puesto que ésta involucrada en las etapas de arquitectura de información, diseño y desarrollo; es un elemento clave en estos procesos porque permite localizar las cuestiones que deben ser revisadas o corregidas en un primer estadio, cuando aún están en temprano desarrollo.

PROTOTIPO DE BAJA FIDELIDAD

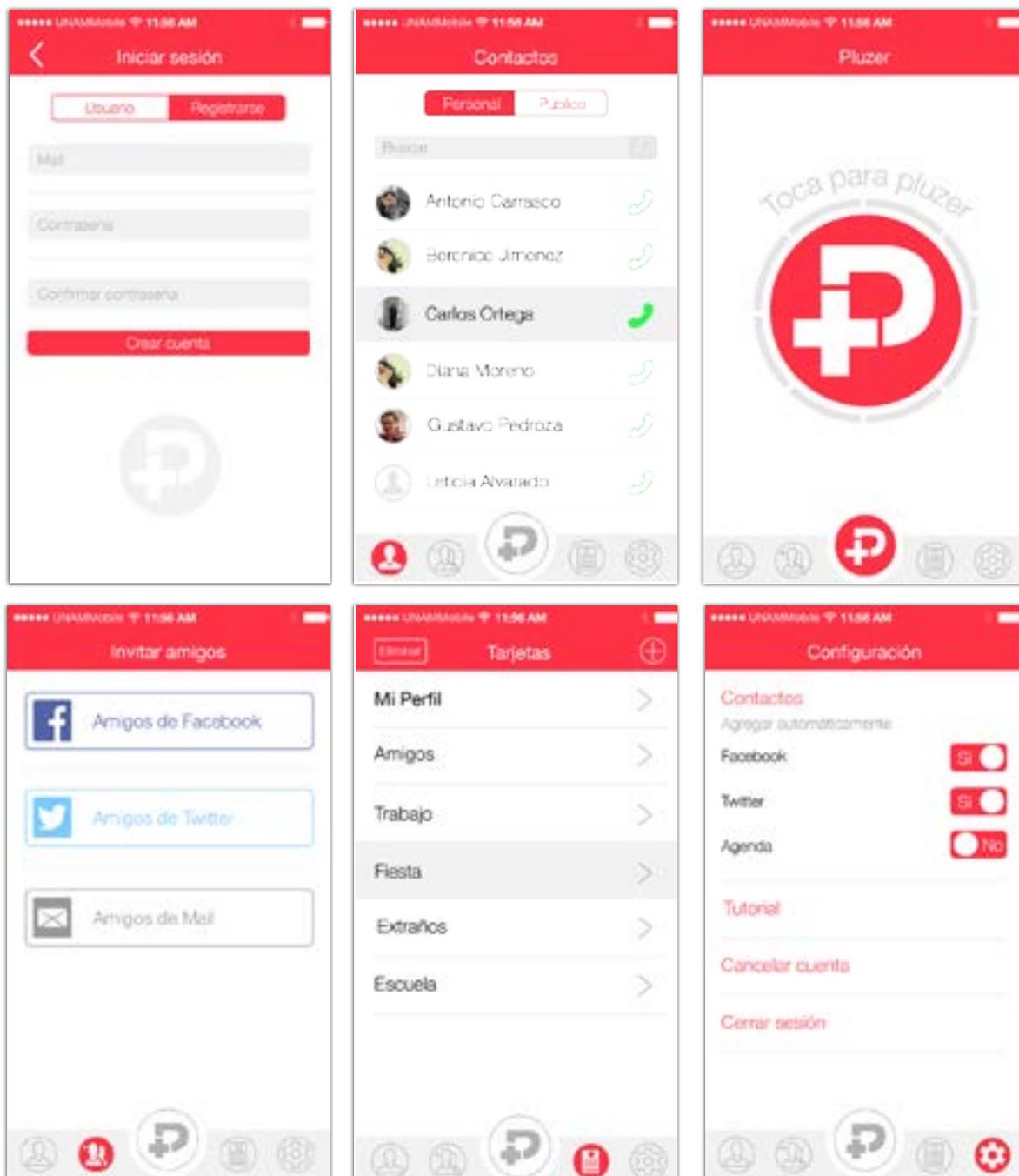
Utilizado en la etapa de arquitectura de la información, son también llamados wireframes, por lo tanto distan del diseño final y pueden realizarse sobre papel u ordenador. Su función es esquematizar las propuestas de estructura de pantalla y aunque parezca muy pronto, permiten realizar los primeros estudios de usabilidad.



WIREFRAMES DE LAS PANTALLAS PRINCIPALES DE PLUZER

PROTOTIPO DE ALTA FIDELIDAD

Utilizados en la parte de diseño y desarrollo, se elaboran por ordenador, y representa un aspecto muy similar al de la aplicación final. Su función es verificar que las necesidades se estén cubriendo y realizar evaluaciones de usabilidad para corregir problemas antes de pasar a niveles avanzados de desarrollo. Como es indicado por Monjo (2011), este tipo de prototipo se divide en dos fases: el prototipo gráfico u horizontal y el prototipo funcional o vertical.



PROTOTIPOS DE ALTA FIDELIDAD PARA LA APP PLUZER

Prototipo gráfico u horizontal. Consiste en la preparación gráfica, en alta fidelidad, de las pantallas clave de la app. En una aplicación normal suelen corresponder al menú principal, a varias pantallas de contenido, eligiendo las más significativas, como exposición de todas las posibilidades, y al tratamiento de otras pantallas especiales que puedan existir.

De esta manera, se obtienen una serie de gráficos que permiten ver si la distribución de los elementos es correcta, y si puede responder realmente a los contenidos previstos.

Prototipo funcional o vertical. Supone la preparación de una versión temprana y reducida de la aplicación, aquí algunas de las pantallas ya están en funcionamiento . El usuario tiene la capacidad de navegar por el sistema mediante las mismas opciones que contendrá la aplicación final, aunque no puede acceder realmente a todos los contenidos.

Puesto que el desarrollo se encuentra en una fase temprana, es importante realizar una evaluación exhaustiva de los aspectos de diseño y experiencia de usuario, ya que las correcciones efectuadas en etapas posteriores suelen resultar problemáticas, llegando a desajustar todas las previsiones de producción, y en casos extremos algunos de los requerimientos suelen ser afectados teniendo que ser modificados.

De igual forma, el prototipo funcional se presenta al cliente, que tiene una primera muestra de lo que va a ser la aplicación, mediante la que puede aportar sugerencias o realizar correcciones que en esta fase del proyecto son fácilmente realizables (p. 180).

4.7 EVALUACIÓN

En cualquier desarrollo de software de cualquier tipo es importante evaluar la calidad del producto con base a la siguientes características:

Funcionalidad. Atributos que satisfacen las necesidades implícitas o explícitas

Fiabilidad. Capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido),

Eficiencia. Nivel de desempeño del software y la cantidad de recursos necesitados bajo condiciones establecidas.

Mantenibilidad. Facilidad de extender, modificar o corregir errores en un sistema software.

Portabilidad. Capacidad de un sistema software para ser transferido desde una plataforma a otra.

Usabilidad. Esfuerzo necesario para su uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

Ésta última característica será la que compete al diseñador gráfico, porque es la más importante dentro del diseño centrado en el usuario; las demás particularidades no es que tengan menor repercusión pero están orientadas sobre todo a la programación del software, la cual es labor del desarrollador.

La usabilidad es la cualidad que tiene una app o cualquier tipo de software para ser usado con facilidad, o como lo dicta la norma internacional ISO 9241: “la usabilidad es la medida en que un producto puede ser utilizado por determinados usuarios para conseguir ciertos objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso definido”. Como es dependiente de los prototipos puede llevarse a cabo en varios momentos, durante el proceso de diseño o de desarrollo como después de los mismos.

Los principales métodos para la evaluación de la usabilidad en consideración de Monjo (2011) son: el paseo cognitivo, el test con usuarios, y la evaluación heurística. La elección de un método u otro depende básicamente del presupuesto reservado a la evaluación, la adecuación al tipo de proyecto y las limitaciones de tiempo.

PASEO COGNITIVO

El paseo cognitivo puede ser realizado en cualquier fase del diseño utilizando un prototipo, un borrador o el producto final. Teniendo en cuenta los objetivos de los usuarios, los evaluadores reúnen un grupo de tareas significativas y las descomponen en los pasos necesarios para realizarlas. A continuación se ejecuta cada tarea, analizando si puede resultar difícil para el usuario identificar y utilizar el elemento de la interfaz más adecuado para su objetivo, y si la respuesta que devuelve el sistema es suficientemente clara.

El paseo cognitivo tiene en cuenta los factores que intervienen en el proceso mental de toma de decisiones por parte del usuario, como la memoria de trabajo y la habilidad para razonar. Este método es muy adecuado para comprobar la usabilidad de un sistema orientado a usuarios con poca o ninguna experiencia, que operan de modo exploratorio para aprender a utilizar la aplicación.

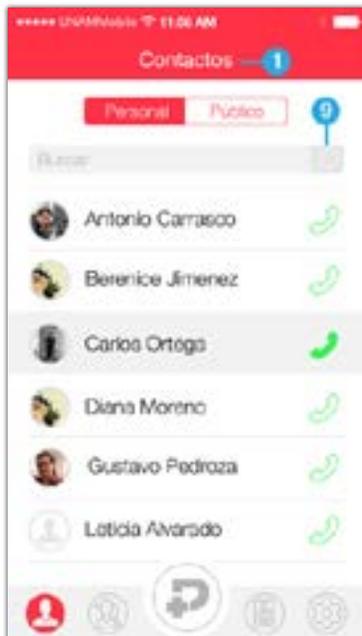
TEST CON USUARIOS

Propuesto por Krug (2006) consultor de usabilidad en Apple, éste es el método más utilizado cuando se quieren “conocer los problemas de usabilidad con los que puede encontrarse el usuario final. Se basa en la observación de los usuarios mientras ejecutan tareas representativas. La repetición del test con varios usuarios permite descubrir qué aspectos del diseño necesitan mejorarse”.

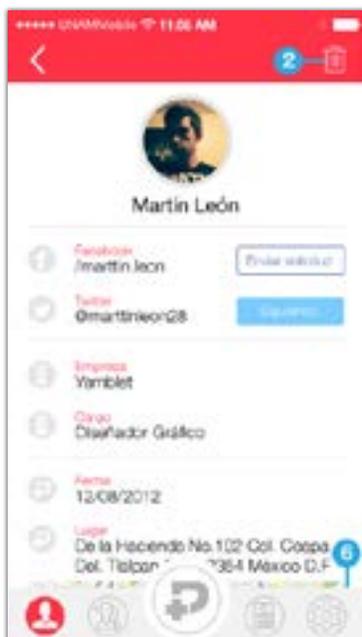


Recomendaciones:

- Que el nombre de la tarjeta en la pantalla pluzer no sea visible para el receptor



PRINCIPIOS 1 Y 9



PRINCIPIOS 2 Y 6

del código QR, que sea visible por unos segundos o que sea visible cuando le dan click.

- Que exista una imagen de contacto.
- Poner fecha y lugar al momento de hacer pluzer.
- Una versión bussiness puede ser generar tarjeta QR para datos de facturación.
- Detectar región para usar el idioma correcto.
- Recomendación de que la primera versión se saque solamente con Twitter y Facebook, porque Google + aún genera problemas en iOS.
- Que sea multicuenta de red social, o sea, que se puedan asociar varias cuentas del mismo tipo de red social.

EVALUACIÓN HEURÍSTICA

La evaluación heurística es un método de inspección, es decir, se realiza por un grupo de expertos en usabilidad, que examinan la interfaz y determinan su grado de cumplimiento de los principios de usabilidad.

El resultado de la evaluación heurística debe ser una lista detallada de cada uno de los problemas de usabilidad hallados en la interfaz, con referencias a los principios que no se han respetado en cada caso, y las opiniones del evaluador.

Los diez principios generales de evaluación heurística presentados por Nielsen (2006) doctor en diseño de interfaces de usuario y ciencias de la computación, son los siguientes: Poner cada pantalla y señalar donde se ubica el principio

1) Visibilidad del estado del sistema. El sistema debe mantener siempre informado al usuario de lo que está ocurriendo, y proporcionarle respuesta en un tiempo razonable.

2) Consistencia entre el sistema y el mundo real. El sistema debe utilizar el lenguaje del usuario, con expresiones que le resulten familiares. Las metáforas son la mejor referencia.

3) Control del usuario. El usuario debe tener la capacidad de abandonar en cualquier momento una situación indeseada o accidental. Asimismo, podrá deshacer o repetir una acción.

4) Consistencia y estándares. El lenguaje utilizado debe ser coherente con las convenciones del sistema operativo.

5) Prevención de errores. Es importante prevenir la existencia de errores; si, a pesar de todo, deben aparecer mensajes de error, éstos han de contener opciones de confirmación antes de ejecutar las acciones de corrección.

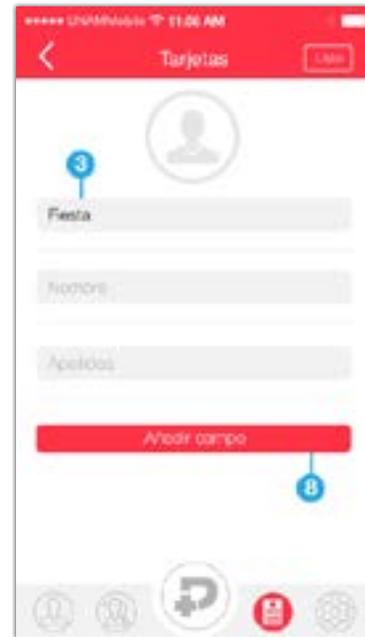
6) Es mejor reconocer que recordar. Para que el usuario no se vea obligado a memorizar continuamente detalles de la navegación, los objetos, acciones y opciones deben estar a la vista. El usuario no tiene que recordar información de una parte de una ventana de diálogo a la siguiente. Las instrucciones de uso o la ayuda del sistema deben estar a la vista, o ser fácilmente accesibles.

7) Flexibilidad y eficiencia de uso. El sistema debe estar preparado para satisfacer tanto a los usuarios novatos, como a los experimentados. Para éstos, resulta muy recomendable incorporar atajos de teclado, que permiten acelerar el proceso de interacción. Los usuarios han de poder configurar sus propios atajos de teclado para acciones frecuentes.

8) Diseño práctico y sencillo. Las pantallas o páginas no deben contener información innecesaria o irrelevante, ya que distrae al usuario y entorpece la navegación. Si aun así es necesario incluir información auxiliar, ésta puede colocarse en páginas distintas, accesibles a través de enlaces.

9) El usuario debe disponer de ayuda para reconocer, diagnosticar y deshacer errores. Los mensajes de error deben presentarse con un lenguaje sencillo, indicando el problema de manera precisa, y sugerir las posibles soluciones.

10) Ayuda y documentación. Aunque es mucho mejor que el usuario pueda navegar sin ayuda, la complejidad de un



PRINCIPIOS 3 Y 8



PRINCIPIOS 4 Y 5



PRINCIPIOS 7 Y 10

aplicaciones para su distribución. Esta es última fase del proceso que si bien no es considerada en los libros, a mi parecer debe ser tomada en cuenta porque es la conexión para que el producto llegue al usuario.

El seguimiento a los comentarios de los usuarios hacia la app en las tiendas es importante porque permite valorar si cubrió las expectativas de éste, además de identificar las virtudes y defectos de la aplicación que deben ser tomados en cuenta para posibles versiones posteriores.

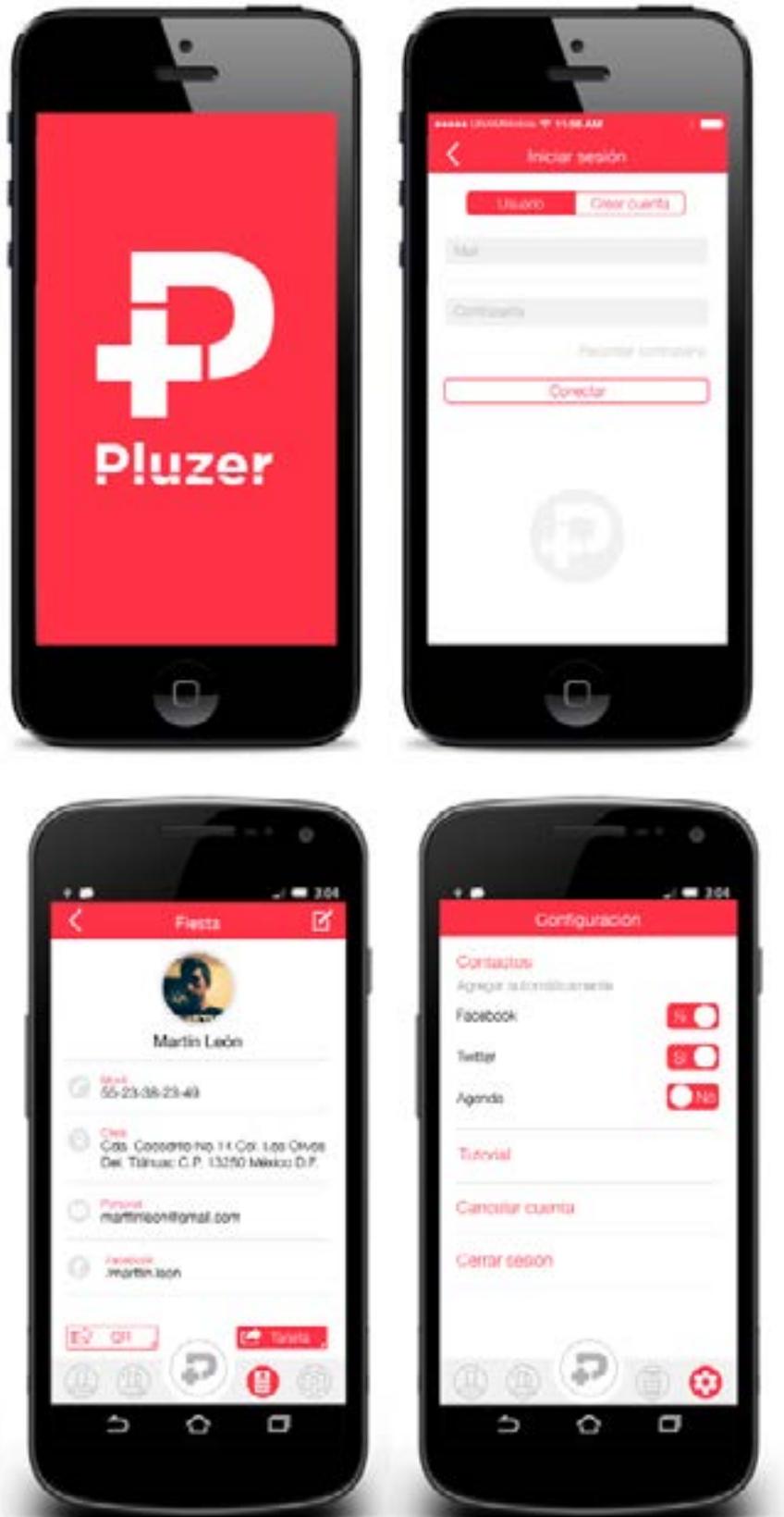
sistema puede recomendar incluir documentación de ayuda. Esta documentación debe ser fácil de encontrar, centrarse en la tarea del usuario, enumerar claramente los pasos que deben llevarse a cabo, y no ser extensa.

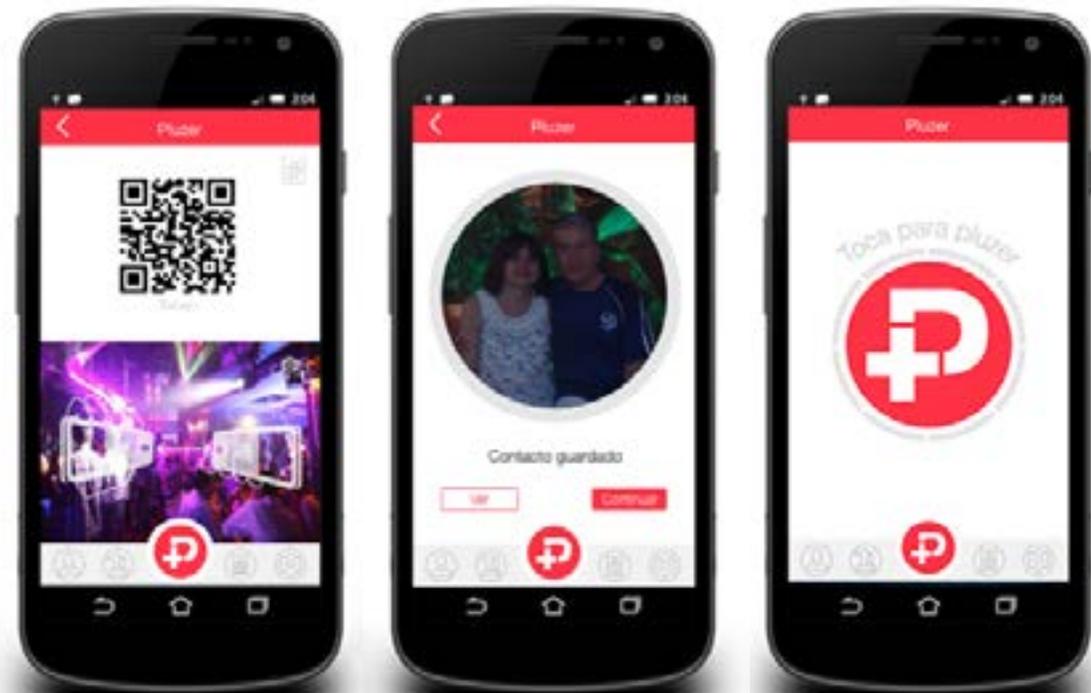
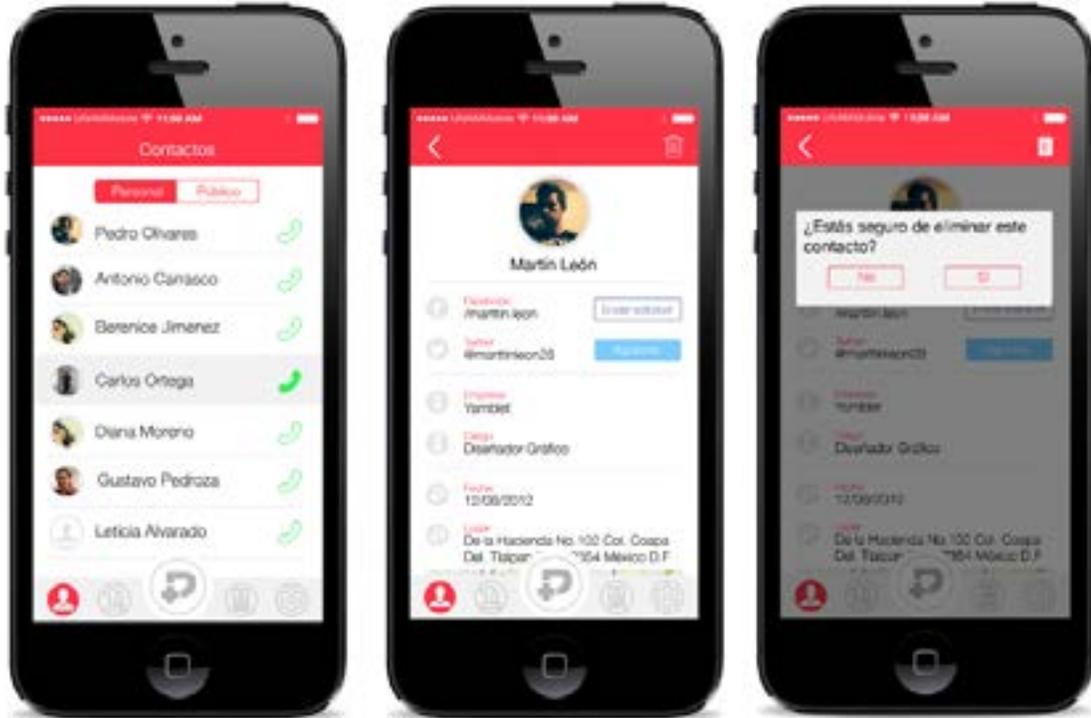
Por último cuando la app ya está terminada entra el equipo de documentación y guionistas, que preparan el contenido y lo estructuran de acuerdo con las características del producto final.

4.8 LANZAMIENTO EN TIENDAS

Cuando todos los errores han sido corregidos por parte del diseñador y del desarrollador, nuevamente se somete a evaluación el prototipo; si los resultados son positivos, este ya puede ser considerado la primer versión de la app y si el cliente da su autorización deberá ser subida a la tienda de

4.9 RESULTADO FINAL





CONCLUSIONES

Cientos de aplicaciones móviles son publicadas diariamente en las distintas App Stores, sin embargo el hecho de estar en tienda no garantiza que vaya a ser exitosa, muchas de ellas pasan al olvido por parte de los usuarios por su mal funcionamiento, poca intuitividad de uso y mal manejo de gráficos, son muy poco atractivas y si llegan a descargarse al instante son desinstaladas; el usuario busca apps cuya calidad vaya más allá de los contenidos, es decir calidad visual. Ésta está ligada con el trabajo del diseñador gráfico por lo que la investigación realizada en esta tesis, fue encaminada hacia la valoración del diseño en la industria móvil.

Sin lugar a dudas la competencia que ha existido entre Apple y Android ha servido para lograr grandes avances en tecnología móvil, tanto en hardware como en software; uno ha provocado al otro incluso a cambiar parte de su filosofía, si Android no hubiera permitido el libre desarrollo en su plataforma difícilmente Apple lo hubiera hecho sobre todo por la exclusividad por la que siempre se le ha caracterizado. De igual forma esta competencia ha motivado a los desarrolladores a expandir su mercado haciendo aplicaciones multiplataforma por lo regular nativas, para aprovechar los distintos sensores de los teléfonos inteligentes, ya que, si bien las web apps son más fáciles de desarrollar poco a poco pierden terreno con las páginas web responsivas y pierden la esencia del Sistema Operativo.

La manera en que un diseñador logra dar solución a los problemas gráficos se manifiesta a través de las metodologías del diseño resultantes del estudio y análisis de la manera de trabajar y pensar del diseñador, la forma en que se constituyen los procesos de diseño, así como la aplicación del conocimiento para la resolución de problemas. Con el conocimiento de las metodologías se reveló el parentesco hacia los métodos de desarrollo de software sobre todo en las metodologías propuestas por Morris Asimow y Munary, en las que en resumen en su fase planeación se busca una necesidad, se estudia que tan factible es solucionarla, si lo es, crear un proyecto preliminar que de una idea de lo que se trata para posteriormente detallarlo; en la fase de producción y consumo es donde entra la planeación de la producción, distribución, consumo y retiro. Además en esta fase intervienen la preparación del diseño, el diseño total de subsistema y componente, diseño detallado de las partes, el dibujo del conjunto, los modelos, las pruebas y análisis y si se da el caso el rediseño.

Para empezar a diseñar el arte de una aplicación móvil es importante entender las características que deben tener los gráficos de una app, entendiéndose que no se trata de diseñar como uno quiera; cada sistema operativo tiene aunque sea de forma muy parecida a las otras, ciertas particularidades que se deben cumplir para el buen uso de la aplicación y permitir una buena interacción entre el dispositivo y el usuario.

Con la elaboración de una app, se conoció a fondo el proceso de desarrollo de software para móviles, en este estudio se encontró cómo el trabajo del diseñador está presente en todo el proceso de desarrollo y se involucra en cada una de las etapas para asegurar un producto de calidad. Su intervención es vista sobre todo en la estructuración de los contenidos de la aplicación y la interacción que existe entre ellos; además, en la parte visual aunque existen muchos gráficos disponibles en la Web, el programador no cuenta con el conocimiento adquirido por el diseñador para el correcto uso estético de estos elementos. Los test de usabilidad han servido como forma de evaluación para calificar el desempeño del diseñador, al lograr o no el fácil uso de la aplicación.

El diseño es una disciplina realmente interesante, llena de posibilidades y caminos que se dividen especialmente a partir de su relación con la tecnología. El diseñador debe ir más allá de ser el constructor de interfaces, en una aplicación podemos hablar tanto de diseño de información, como de interacción o visual; conocer cómo se idean y ejecutan otras etapas del diseño y desarrollo de una app. De esta forma, la relación e integración con otros miembros del equipo puede ser más fructífera, al generar sinergia que resulte en aplicaciones de mayor calidad.

Un diseñador debería entonces, en mayor o menor medida, involucrarse en todas las etapas del proceso, incluso en aquellas que le resulten más técnicas o tediosas. Esto no es algo que deba evitarse, por el contrario, debería aprovecharse para trasladar el aprendizaje al trabajo propio. De la misma manera que los diseñadores deben dar un paso más allá, también los desarrolladores deberían entender el trabajo de diseño y ser conscientes del impacto que tiene en los usuarios una interfaz que, además de verse bien, cuida aspectos de usabilidad. Esta relación puede permitir al programador incorporar una visión estética y de sentido visual, que le ayude a tomar decisiones a la hora de construir una aplicación, guiado también por su instinto y sin depender enteramente del diseñador.

Por medio de ésta investigación y por los dos años de experiencia que llevo trabajando con dispositivos móviles, debo decir que tanto diseñadores como desarrolladores deberían trabajar de la mano en todo momento, para lograr aplicaciones que cuiden todos y cada uno de los detalles. Después de todo, la primera impresión es la que cuenta y el éxito de una aplicación móvil esta determinado por diversos factores: funcionalidad, eficiencia, interés por los contenidos, promoción, experiencia de usuario, usabilidad, calidad artística, estos últimos correspondiendo a lo hecho por el diseñador gráfico; sin embargo existe una condición ineludible: el usuario debe poder alcanzar fácil y cómodamente sus objetivos.

Diseñar aplicaciones no es una tarea fácil, especialmente si no se ha trabajado antes con ellas; para los diseñadores es todo un desafío empezar a diseñar para móviles, pero también, es una buena oportunidad para meterse en este nuevo ámbito, se trata de un nuevo soporte en donde aplicar el diseño y que poco a poco a tomado mayor fuerza. En la Web ya existen páginas especializadas en este tema con recursos que sirven de inspiración y no sería mala idea que en las escuelas de diseño gráfico se comenzará a enseñar.

FUENTES DE CONSULTA:

Alasdair, A. (2013). *Learning iOS Programming* (3ra ed.). California: O'Reilly Media Inc.

Android (2013). *Android Design*. Estados Unidos: Android Open Source Project. <http://developer.android.com/design/index.html>

Ankeny, J.(2011) Diez mil millones de descargas y contando: La historia de la App Store de Apple, y sus aplicaciones de todos los tiempos mejores. FierceDeveloper. <http://www.fiercedeveloper.com/special--reports/ten--billion--downloads--and--counting--history--apples--app--store--and--its--all--time>

Apple (2013). *iOS Human Interface Guidelines*. Estados Unidos: Apple Inc. https://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobihig/index.html#//apple_ref/doc/uid/TP40006556-CH66-SW1

Asimow, M. (1912). *Introduction to design*. USA: Prentice-Hall Inc, 1962, págs. 1-17.

Bochenski, I. (1969). *Los métodos actuales del pensamiento*. Madrid: Ediciones Rialp.

Bonsiepe, G. (1978). *Teoría y prácticas del diseño industrial*. Barcelona: Ed. Gustavo Gili, 254 págs.

Cooper, A. (2011). *Newsletters on personas*. San Francisco, CA.: Cooper. <http://www.cooper.com/content/insights/newsletters_personas.asp>

Cross, N. (1996). *Métodos de diseño: estrategias para el diseño de productos*. Editorial Limusa.

Cuello, J., Vittone, J. (2013). *Diseñando apps para móviles*. Barcelona. <http://www.appdesignbook.com/es/contenidos/disenio-visual/>

Doncel, M. (2012). *¿Qué es una web app?*. España: STARTCAPPS. <http://www.startcapps.com/blog/que-es-una-web-app>

Felker, D. (2011). *Android Applications Development for Dummies*. Indianapolis: Wiley Publishing Inc.

Ferrer, A. (2011). *Escritura y tipografía*. España: Editorial Universitat Obertat de Catalunya.

- Frascara, J. (1998). *Diseño Gráfico y Comunicación*. Ed Infinito, 127 págs.
- Goldstein, M., Wilson, D. (2013). *iOS 6 Applications Development for Dummies*. New Jersey: John Willey & Sons, Inc.
- Hassan, M., Fernández M. (2004). *Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la información*. Barcelona. <<http://www.hipertext.net/web/pag206.htm#Modelado%20del%20usuario>>
- Jones, C. (1996). *Métodos de diseño*. Barcelona: Ed. Gustavo Gili, 370 págs.
- Koetsier, J. (2013). *Android up 13%, iOS down 7%, BlackBerry down 81% ... and Windows Phone up a massive 52%*. Venturebeat. <http://venturebeat.com/2013/04/01/android-up-13-ios-down-7-blackberry-down-81-and-windows-phone-up-a-massive-52/>
- Krug, S. (2006). *No me hagas pensar*. Madrid: Editorial Pearson Educación.
- Löbach, B. (1981). *Diseño industrial*. Barcelona: Ed. Gustavo Gili, 204 págs.
- Monjo, T. (2011). *Diseño de Interfaces Multimedia*. España: Editorial Universitat Obertat de Catalunya.
- Munari, B. (1983). *¿Cómo nacen los objetos?* .Barcelona: Ed. Gustavo Gili. 385 págs
- Nielsen, J., Loranger, N. (2006). *Usabilidad. Prioridad en el diseño web*. Madrid: Editorial Anaya Multimedia.
- Ostrander, J. (2012). *Android UI Fundamentals: Develops and Design*. California: Peachpit Press.
- Papanek, V. (1984). *Diseño para el mundo real*. Londres: Ed. Thames & Hudson,
- Prieto, J., Rámirez R. (2011). *Tecnología y Desarrollo en dispositivos móviles*. España: Editorial Universitat Obertat de Catalunya.
- Ratabouil, S. (2012). *Android NDK Beginner's Guide*. Birmingham: Pack Publishing Ltd.
- Rouse, M. (2013). *Search Software Quality*. Massachusetts: TechTarget. <http://searchsoftwarequality.techtarget.com/definition/native-application-native-app>

Singh, S. (2013). *Understanding different types of Mobile applications*. Estados Unidos: exploreB2B. <https://exploreb2b.com/articles/understanding-different-types-of-mobile-applications>.

Taylor B. (2013). *Starter's Guide to iOS Design*. Estados Unidos. <http://taybenlor.com/2013/05/21/designing-for-ios.html>

Vilchis, L. (2002). *Metodología del diseño, fundamentos teóricos*. México: Ed. Claves Latinoamericanas, 161 págs.

Wikipedia (2013). *Sistema hexadecimal*. Estados Unidos. http://es.wikipedia.org/wiki/Sistema_hexadecimal