

## چالش‌ها و بهترین شیوه‌ها برای توسعه اپلیکیشن موبایل:

### یک مقاله مروری

#### چکیده

در طی ده سال گذشته، فناوری دستگاه‌های موبایل به صورت قابل توجهی تغییر کرده است، و این دستگاه‌ها و سیستم عامل‌های آن‌ها پیچیده تر شدند. این توسعه‌ها به تغییرات بزرگی در اپلیکیشن‌های موبایلی طراحی شده متناسب با سیستم عامل‌های موبایل منجر شدند. طراحی اپلیکیشن‌های موبایل به چند دلیل مانند اندازه صفحه نمایش و قدرت پردازش محدود و سایر موارد، سخت تر شده است. بنابراین، مهم است که به وضوح مشخصه‌های توسعه اپلیکیشن موبایل و مسائل و چالش‌های مربوط به آن، و همچنین، ویژگی‌های کلیدی که یک اپلیکیشن موبایل را سودمند و ارزشمند می‌سازد را به وضوح شناسایی کنیم. این مقاله ادبیات موضوعی موجود در مورد چالش‌ها و بهترین شیوه‌های توسعه اپلیکیشن موبایل را بررسی کرده است. این مطالعه در درک مشخصه‌های فرایند توسعه اپلیکیشن موبایل، بررسی چالش‌های واقعی که در طی این فرآیند دیده می‌شود و اکتشاف بهترین شیوه‌هایی که می‌توانند به صورت فعالی برای بهبود توسعه اپلیکیشن‌های موبایل استفاده شوند، کمک می‌کند.

مفاهیم CCS: رایانش انسان محور، رایانش سیار، رایانش انسان محور، مطالعات تجاری در رایانش سیار و فراگیر

**کلمات کلیدی:** اپلیکیشن موبایل، مهندسی نرم افزار موبایل، توسعه موبایل، بهترین شیوه‌ها، چالش‌ها

## 1. مقدمه

همانطور که دیده شده است، در دهه گذشته، صنعت ارتباطات مخابراتی سیار یکی از بخش‌های دیجیتال با رشد سریع بوده است که یک محیط به شدت رقابتی، غیرقطعی و پویا را بوجود آورده است [3]. این زمینه برای سازمان‌های نرم افزاری به دلیل درآمد قابل توجهی که می‌توانند از آن کسب کنند، به یک زمینه بسیار جالب تبدیل شده است [3]. علاوه بر گسترش آن، برای جوامع پژوهشی و صنعت نرم افزاری مسائل و نگرانی‌هایی را به همراه داشته است. برای نمونه، فرآیند توسعه اپلیکیشن موبایل کاملاً از نرم افزار سنتی متفاوت است، چرا که کاربران نهایی ویژگی‌هایی مشابه با اپلیکیشن‌های کامپیوترهای دسکتاپ با ویژگی‌هایی منحصر تلفن همراه را انتظار دارند [11].

علاوه بر این، به منظور ایجاد یک اپلیکیشن موبایلی خوب، در بر گرفتن ویژگی‌های اصلی که یک اپلیکیشن موبایل خوب را مشخص می‌کند، که اگر عملی شوند، آن‌ها را ارزشمند و سودمند می‌سازد، ضروری است. بنابراین، هدف مطالعه ما کسب درک بهتر از مشخصه‌های توسعه اپلیکیشن موبایل و چالش‌های متفاوت و مسائل مرتبط به آن است. به همین دلیل، مطالعه مروری فعلی پاسخ به دو سوال را هدف قرار داده است: (1) فرآیند توسعه اپلیکیشن‌های موبایل با چه چالش‌ها و مسائلی مواجه می‌شوند؟ و (2) بهترین شیوه‌ای که معمولاً برای برطرف کردن چالش‌های فرآیند توسعه اپلیکیشن موبایل استفاده می‌کنیم کدام است؟

این مقاله تلاشی برای بررسی ادبیات موضوعی منتشر شده در مورد چالش‌ها و بهترین شیوه‌هایی است که می‌تواند در طول توسعه اپلیکیشن‌های موبایل تحقق بخشیده شود. مطالعه مروری در درک بهتر مشخصه‌های فرآیند توسعه اپلیکیشن موبایل، بررسی چالش‌های واقعی که با آن‌ها مواجه می‌شوند و کشف بهترین شیوه‌هایی که می‌توانند به صورت موثری برای بهبود توسعه اپلیکیشن موبایل استفاده شوند نقش دارد.

مقاله به شرح زیر سازمان یافته است. بخش 2 یک بررسی بر مهندسی نرم افزار برای توسعه اپلیکیشن‌های موبایل ارائه می‌دهد. بخش 3 فرآیند توسعه کلی اپلیکیشن موبایل را تشریح می‌کند. بخش 4 مشخصه‌های اپلیکیشن‌های موبایل را شناسایی می‌کند. بخش 5 مسائل و چالش‌های مربوط به فرآیند توسعه اپلیکیشن موبایل را که توسعه دهندگان موبایل

با آن‌ها روبرو می‌شوند را دسته بندی و شناسایی می‌کند. سپس، بخش 6 بهترین روش توسعه اپلیکیشن موبایل را مشخص می‌کند. سرانجام، بخش 7 نتیجه گیری مقاله و طرحی برای کارهای آینده است.

## 2. مهندسی نرم افزار برای توسعه اپلیکیشن‌های موبایل

در سال‌های گذشته، صنعت نرم افزار شاهد رشد سریعی بوده است و تعداد بالقوه اپلیکیشن‌های موبایل نامحدود شده‌اند. با افزایش شهرت و تقاضا برای اپلیکیشن‌های موبایل، تعداد پروژه‌های توسعه برای اپلیکیشن موبایل به صورت قابل توجهی افزایش یافته است [8]. در نتیجه، کیفیت و کمیت اپلیکیشن‌های موبایل نگرانی‌های جدیدی را در صنعت مهندسی نرم افزار بوجود آورده‌اند. توسعه اپلیکیشن موبایل یک فرآیند توسعه اپلیکیشن برای دستگاه‌های کوچک، کم توان مانند تلفن‌های سلولی است [9][8]

این اپلیکیشن می‌تواند یک اپلیکیشن از قبل نصب شده کارخانه ای یا قابل دانلود از فروشگاه‌های نرم افزاری (App-stores) یا پلت فرم‌های توزیع نرم افزار موبایل باشد [8]. در حال حاضر، اپلیکیشن‌های بیشماری در app stores برای کمک به زندگی بر اساس تلفن همراه، یا mLife در دسترس هستند که شامل: mEnvironment, mEducation, mTourism, mGovernment, mEntertainment می‌باشد. اپلیکیشن‌های موبایل می‌توانند به موارد زیر دسته بندی شوند [23] [17]:

- اپلیکیشن‌های بومی : این اپلیکیشن‌ها برای یک پلت فرم اصلی با استفاده از کیت‌های توسعه نرم افزار (SDK) و ابزارهای ارائه شده توسط مالک پلت فرم اجرا شده اند، و این ممکن است تعداد دستگاه‌های موبایلی که اپلیکیشن را اجرا می‌کنند را محدود کنند اما این می‌تواند از همه ویژگی‌های پلت فرم که توسط توسعه دهندگان ارائه شده است، بهره مند شود.

- اپلیکیشن‌های وب : این اپلیکیشن‌ها با استفاده از تکنولوژی‌های وب استاندارد (مانند HTML, CSS و JavaScript) اجرا شده اند و می‌توانند از طریق مرورگر وب موبایل در دسترس قرار گیرند. این در دسترس بودن

اپلیکیشن در اینترنت را برای پلت فرم‌های موبایل متفاوت تضمین می‌کند اما این اپلیکیشن‌ها نمی‌توانند از همه قابلیت‌های موبایل استفاده کنند و نمی‌توانند به صورت آفلاین استفاده شوند.

• اپلیکیشن‌های هیبریدی: این اپلیکیشن‌ها مزایای اپلیکیشن‌های بومی و وبی را با یکدیگر ترکیب می‌کند.

### 3. فرآیند توسعه اپلیکیشن موبایل

توسعه اپلیکیشن موبایل یک مورد خاص توسعه نرم افزار است چرا که توسعه دهندگان نیاز دارند که جنبه‌های متفاوتی مانند طول عمر کوتاه توسعه، قابلیت‌های دستگاه موبایل، قابلیت حرکت، قابلیت دیده شدن، مشخصات دستگاه موبایل مانند اندازه صفحه نمایش، طراحی و ناوبری واسط کاربر اپلیکیشن، امنیت و حریم شخصی کاربر، ضرورت بازاریابی اپلیکیشن برای کسب شهرت بیشتر را در نظر بگیرند [7].

در کل، طول عمر توسعه اپلیکیشن‌های موبایل از موارد زیر تشکیل شده است: (1) تحلیل ایده اپلیکیشن؛ (2) طراحی واسط کاربر؛ (3) توسعه اپلیکیشن با استفاده از ابزارها و زبان‌های برنامه نویسی برای پلت فرم هدف؛ (4) تست اپلیکیشن در دستگاه‌های گوناگون؛ (5) انتشار اپلیکیشن در فروشگاه پلت فرم هدف، و در آخر (6) به روز نگه داشتن اپلیکیشن با نسخه‌های جدید منتشر شده در فروشگاه‌های اپلیکیشن [7].

توسعه اپلیکیشن موبایل با توجه به (1) محدودیت‌های فنی مربوط به سیستم عامل موبایل، و (2) ماهیت تغییرات سریع پیش نیازهای تجاری، یک وظیفه چالش برانگیز دیده شده است [8]. همچنین این چالش وجود دارد که تیم‌های توسعه برای کار در یک محیط پویا با تغییرات مکرر نیازهای مشتریان و انتظارات آن‌ها، مقابله می‌کنند [8]. چالش‌های توسعه اپلیکیشن موبایل به صورت مفصل در بخش 5 مورد بحث قرار گرفته است. با توجه به این حقیقت که فرآیند توسعه اپلیکیشن موبایل مشابه با مهندسی نرم افزار است، پیش نیازهای اضافی را معرفی می‌کند که تنظیماتی از فرآیند توسعه نرم افزار سنتی را تحمیل می‌کند [9, 29, 31]. این پیش نیازهای خاص و مشخصه‌ها به صورت دقیق در بخش بعدی بحث شده اند.

#### 4. مشخصه‌های اپلیکیشن‌های موبایل

به منظور توسعه اپلیکیشن‌های موبایل با کیفیت بالا، تشخیص مشخصه‌های کلیدی که کیفیت اپلیکیشن‌های موبایل را تعریف می‌کنند، ضروری است [10]. مشخصه کیفیت که برای توسعه همه اپلیکیشن‌های نرم افزاری در نظر گرفته شده اند را "عوامل بهره وری" گویند (مانند پیش نیازهای عملیاتی، پیش نیاز قابلیت استفاده، پیش نیاز قابلیت اعتماد، پیش نیاز قابل حمل بودن، پیش نیازهای قابلیت نگهداشت، و پیش نیازهای کارامدی) که بین اپلیکیشن‌های نرم افزاری و موبایلی مشترک هستند [17]. به هر حال، مشخصه‌های خاصی برای توسعه اپلیکیشن موبایل برای دیگر فاکتورهای بهره وری مشخص شده اند [17]. این بخش شناخت این مشخصه‌ها که نشان می‌دهند که چقدر اپلیکیشن‌های موبایل از نرم افزار سنتی متفاوت هستند را هدف قرار می‌دهد.

Ashishdeep و همکاران (2016) [2] مشخصه‌های اصلی اپلیکیشن موبایل را با توجه به برخی از جنبه‌ها شناسایی کردند: پیش نیازهای کاربر در طول زمان دائما تغییر می‌کنند، یک درخواست تغییر نیاز است، که بر مولفه‌های UI تاکید دارد، و زمان توسعه/استقرار باید کوتاه باشد. از بررسی ادبیات موضوعی، آن‌ها بیشتر توجهات خود را به رویکردهای چابک<sup>1</sup> معطوف کردند که بهترین رویکرد مناسب را برای توسعه اپلیکیشن‌های موبایل در نظر می‌گیرد.

Kaleel and Harishankar (2013) [15] تاکید کردند که پیش نیاز اپلیکیشن موبایل مستقیما به تقاضای درخواست کاربر نهایی بستگی دارد، و آن‌ها اشاره کردند که صنعت اپلیکیشن موبایل بیشتر بر راه اندازی محصول متمرکز است تا پایبندی به فرآیند چرخه حیات توسعه نرم افزار<sup>2</sup> (SDLC)، چرا که صنعت موبایل منعکس کننده دیدگاه کاربران نهایی است و اپلیکیشن به سرعت برحسب تقاضا نصب می‌شود. آن‌ها بیشتر به اهمیت آگاهی بخشی بازار در صنعت اپلیکیشن موبایل اشاره می‌کنند که برای حفظ حداقل زمان برای نصب اپلیکیشن تلاش می‌کنند.

De Souza و De Aquino (2014) [25, 26] مشخصه‌های اپلیکیشن موبایل را با انجام یک بررسی سیستماتیک شناسایی کردند. این مشخصه‌ها در زیر تشریح شدند:

<sup>1</sup> Agile approaches

<sup>2</sup> Software Development Life Cycle

- انرژی محدود: هر دستگاه موبایل یک باتری با یک طول عمر محدود دارد، لذا اجرای اپلیکیشن نباید نیازمند مقدار زیادی از منابع سخت افزاری باشد.
- واسط گرافیکی: طراحی واسط محدود است چرا که اندازه صفحه نمایش کوچک است.
- واسط ورودی: با ابزارهای تعاملی بین کاربران و اپلیکیشن مانند صفحه کلید، صفحه لمسی یا صوتی و تشخیص تصویر شناخته می شود.
- پهنای باند: یک اپلیکیشن موبایل باید تغییرات گسترده پهنای باند را در نظر بگیرد و متوجه شود، چرا که ممکن است برخی از مواقع حداکثر پهنای باند را داشته باشد و در شرایط دیگر حداقل پهنای باند را داشته باشد.
- نوع ارتباطات: اپلیکیشن های موبایل باید از انواع گوناگون ارتباطات مانند، بلوتوث، G, 4G, Wi-Fi3، بی سیم و غیره حمایت کنند.
- تغییر در زمینه: کاربران داده های ورودی را به صورت صریح ارائه می دهند، در حالی که اپلیکیشن به صورت ضمنی ورودی هایی را با توجه به زمینه فیزیکی و محاسباتی ارائه می دهند.
- وقفه مداوم در فعالیت ها: برخی از اپلیکیشن ها برای کار به صورت آفلاین طراحی شده اند و در زمانی که اپلیکیشن ها آفلاین می شوند همگام سازی صورت می گیرد. بنابراین، اپلیکیشن های موبایل باید برای سناریوهای متفاوتی طراحی شوند چرا که وقفه در فعالیت های گوناگون مانند باتری ضعیف یا فقدان ارتباط اینترنتی ممکن است رخ دهد.
- عملکرد محدود: همه دستگاه های موبایل محدودیت هایی در منابع اصلی مانند منابع پردازشی، حافظه و ارتباطات دارند.
- حافظه کاهش یافته: به دلیل اندازه محدود سخت افزار، حافظه کمتری در دسترس است.
- زمان پاسخ: توسعه اپلیکیشن باید بر متغیر زمان متمرکز باشد. این مسئله مستلزم این است که اپلیکیشن ها با یک بهینه سازی ممکن منابع برای یک زمان پاسخ و کارآمدی بهتر توسعه یابند.
- پردازش: زمان پاسخ مستقیماً با قدرت پردازش در ارتباط است.

• قابلیت حمل: قابلیت حمل سخت افزار، یک اپلیکیشن موبایل باید از طیف گسترده‌ای از دستگاه‌ها پشتیبانی کند. قابلیت حمل نرم افزار، یک اپلیکیشن موبایل باید ظرفیت اجرای همان چیزها را در طیف گسترده ای از سیستم عامل‌های موبایل داشته باشد.

• اپلیکیشن‌های بومی، وب یا هیبریدی: بر اساس انواع اپلیکیشن، مسائل و پیچیدگی که ممکن است در نظر گرفته شوند، متمایز خواهند بود، و خواستار سطح متفاوتی از تلاش‌های توسعه هستند.

Flora and Chande (2013) [11] بر جنبه‌هایی بحث کردند که معمولا در اپلیکیشن‌های نرم افزار سنتی دیده نمی‌شود، مانند: ادغام اینترفیس اپلیکیشن با اپلیکیشن‌های متفاوت، بررسی حرکت دستگاه مانند اپلیکیشن GPS، سازگاری اپلیکیشن با همه پلت فرم‌ها و فاکتورهای سخت افزاری متفاوت، مدیریت مصرف برق، ریسک‌های امنیتی و حریم شخصی، طراحی UI، پیچیدگی تست، در دسترس بودن بیست و چهار ساعته در هفت روز هفته "همیشه روشن بودن"؛ و برنامه ریزی برای فرآیند توسعه، که شامل اندازه اپلیکیشن مانند اندازه صفحه نمایش اپلیکیشن است. آن‌ها بیشتر بر چالش‌هایی تمرکز کردند که تیم‌های توسعه اپلیکیشن موبایل با آن روبرو می‌شوند مانند محیط پویایی از پیش نیازهای فنی و بهبودهای گوناگون، با تنظیم ثابت در انتظارات و نیازهای کلاینت‌ها. آن‌ها فرآیندهای چابک را به عنوان مناسب ترین مورد برای این محیط‌های دائما در حال تغییر در نظر گرفتند؛ در حالی که رضایت مشتری در مراحل اولیه و غالبا در زمان تحویل مدیریت می‌شود، و همیشه تغییرات جدیدی؛ چرخه انتشار کوتاه، یک همکاری مستمر بین مشتری یا ذی نفع و مهندس خوشایند است، و داشتن یک برنامه کاری یک معیار لازم پیشرفت است.

Kakkar و همکاران (2013) [14] بر تفاوت بین اپلیکیشن‌های موبایل و اپلیکیشن‌های نرم افزاری سنتی بحث کردند، که در آن مسائل اصلی که باید در توسعه اپلیکیشن موبایل مدیریت شود امنیت، قابلیت اعتماد، عملکرد و فضای ذخیره حافظه است. و دیگر مسائلی که با اپلیکیشن‌های موبایل در ارتباط هستند عبارتند از: (1) تعامل با اپلیکیشن‌های متفاوت؛ (2) اپلیکیشن‌های بومی یا ترکیبی و (3) مدیریت حسگر، (4) قابلیت‌های سخت افزاری و نرم افزاری، (59) امنیت، (6) واسط کاربر باید از راهنماهای UI ارائه شده توسط SDK تبعیت کند، (7) پیچیدگی تست و (8) مصرف برق.

Vithani and Kumar (2014) [28] تفاوت‌هایی بین یک کامپیوتر شخصی (PC) و توسعه اپلیکیشن موبایل را

که در جدول 1 خلاصه شده است را شناسایی می‌کند.

جدول 1 تفاوت بین اپلیکیشن‌های موبایل و PC

معیار	اپلیکیشن موبایل	اپلیکیشن PC
طول عمر	کوتاه تر از اپلیکیشن PC	طولانی تر از اپلیکیشن موبایل
ویژگی‌ها	پیچیده، یک یا چند ویژگی (محدود)	اغلب قابلیت‌های ساده، بیشتر
واسط فیزیکی	چند واسط فیزیکی مانند پنل لمسی یا صفحه کلید موبایل	تعداد زیادی واسط فیزیکی مانند صفحه کلید، موس، صفحه لمسی، و دیگر دستگاه‌های خارجی
تعداد صفحه نمایش برای تعامل	صفحه نمایش بسیار برای تعامل، صفحه نمایش کوچک	تعداد صفحه نمایش کمتر برای تعامل، صفحه نمایش بزرگ
توسعه پلت فرم	محدودیت‌هایی در دستاوردهای اپلیکیشن در توسعه اپلیکیشن موبایل در یک پل فرم (بومی) یا توسعه اپلیکیشن در چند پلت فرم	بدون محدودیت با توجه به پلت فرم
استفاده از باتری و حافظه	باید برای استفاده از حافظه موجود به استفاده از باتری و حافظه صورت بهینه طراحی شود.	این یک تصمیم طراحی اصلی نیست چرا که پردازش‌های بیهوده ای در PC اجرا می‌شوند که در مقایسه با اپلیکیشن موبایل خیلی از باتری یا حافظه PC استفاده نمی‌کنند.
به روزرسانی	اجباری	اختیاری

همانطور که قبلاً بحث شد، مطالعات بسیاری وجود دارند که رویکرد چابک را به عنوان یکی از بهترین موردها برای

SDLC اپلیکیشن موبایل می‌بینند. در مقابل، Popa (2013) [23] بیان کرد که برای اپلیکیشن‌های موبایل نیاز

نیست که فازهای SDLC کاملاً تعریف شوند چرا که هر فازی که بتواند قبل از مرحله قبلی شروع شود می‌تواند در



هر زمانی که می‌خواهد خاتمه یابد، که ممکن است بر چرخه کوتاه توسعه و کاهش هزینه توسعه تاکید داشته باشد. همچنین، SDLC مانند آبخاری<sup>۲</sup>، چابک<sup>۴</sup>، اسپیرال<sup>۵</sup>، و غیره می‌توانند برای توسعه اپلیکیشن‌های موبایل استفاده شوند. جنبه‌های کلیدی SDLC اپلیکیشن موبایل به شرح زیر است:

- تجربه کاربر (UX) باید با توسعه نمونه‌های UI در چند پلت فرم ارتقا یابد.
  - تست باید با استفاده از دستگاه‌های گوناگونی انجام شود که دامنه گسترده‌ای از ویژگی‌های سخت افزاری و نرم افزاری را دارد. همچنین، تست اپلیکیشن تنها بر شبیه سازها کافی نیست.
  - نگهداری باید با به روز رسانی اپلیکیشن به آخرین نسخه پلت فرم موبایل انجام شود.
- علاوه بر این، بسیاری از محققین با توجه به استفاده از یک متدولوژی خاص بیان کردند که، جنبه‌های بسیاری وجود دارند که بر انتخاب شیوه‌های متدولوژی برای توسعه اپلیکیشن‌های موبایل تاثیر می‌گذارد، و شناسایی یک متدولوژی یا مجموعه‌ای از روش‌هایی که به صورت کلی به بهترین نحو بر همه پروژه‌های توسعه اپلیکیشن موبایل کار می‌کنند به شدت سخت و گاهی غیرممکن است [8]. از طرفی، رویکردهای چابک در مورد افراد است، نه تکنولوژی و سازمان باید از آنچه که مناسب تیم و فرهنگ خود است استفاده کند، چرا که یک پلت فرم به سختی متدولوژی را تشخیص می‌دهد [8].

Inukollu و همکاران (2014) [12] بر فاکتورهایی بحث کردند که باعث کیفیت پایین اپلیکیشن‌ها می‌شود و دلایل این کیفیت را از نقطه نظر SDLC توضیح دادند که عبارتند از:

- توسعه دهندگان فازهای SDLC را تایید نمی‌کنند، چرا که عمده توسعه دهندگان فاز اجرا را بدون جمع آوری پیش نیازها یا داشتن طرح آغاز می‌کنند.
- فقدان آموزش و تجربه در توسعه SDKها
- پوشش تست کافی نیست.

---

<sup>3</sup> Waterfall

<sup>4</sup> Agile

<sup>5</sup> Spiral

• پشتیبانی و نگهداری ضعیف.

Anwar (2015) [1] گزارش داد که مهم ترین مشخصه‌های کیفی بین توسعه دهندگان اپلیکیشن موبایل با رتبه بالا و پایین، تست اپلیکیشن، تست قابلیت استفاده و تست کاربردی بوده است.

به طور خلاصه، واضح است که توسعه اپلیکیشن موبایل به دلیل محیط خاص آن، محیط محدود، پیش نیازهای دائم در حال تغییر، بازار و سایر موارد کاملاً از توسعه نرم افزار سنتی متفاوت است [4]. علاوه بر این، برای ایجاد یک اپلیکیشن خوب، برای توسعه دهندگان در برگرفتن مشخصه‌های کلیدی که یک اپلیکیشن موبایل خوب را تعریف می‌کنند مهم است [10].

## 5. چالش‌های توسعه اپلیکیشن موبایل

تا امروزه، مطالعات گوناگونی چالش‌های اصلی در رایانش موبایل را شناسایی و منتشر کرده اند. چند مطالعه بر آن‌ها به صورت خلاصه در زیر آورده شده اند.

Wasserman [29] مسائل مرتبط با توسعه اپلیکیشن موبایل را، بر اساس ابزارهای توسعه آن، فرآیندها، قابلیت حمل اپلیکیشن، طراحی واسط کاربر، کیفیت و امنیت شناسایی کرده اند که عبارتند از: (1) تعامل بالقوه با دیگر اپلیکیشن‌ها، (2) مدیریت حسگر، (3) انواع اپلیکیشن‌ها (بومی و هیبریدی مانند وب موبایل)، (4) خانواده‌های پلت فرم‌های سخت افزاری و نرم افزاری پشتیبان OS (5) امنیت، (6) واسط کاربری و محدودیت‌هایی برای دنبال کردن راهنماهای واسط کاربری توسعه یافته مانند SDKها، (7) پیچیدگی تست اپلیکیشن‌های وب موبایل و (8) مصرف برق توسط اپلیکیشن‌های موبایل.

Dehlinger and Dixon [5] چهار چالش اصلی را برای توسعه اپلیکیشن موبایل شناسایی کرده اند که (1) ایجاد یک واسط کاربری سراسری است که چالش‌هایی را با خود به همراه دارد چرا که هر پلت فرم موبایل یک راهنمای منحصر برای پیش نیازهای واسط کاربر دارد، (2) ایجاد توانایی در استفاده مجدد نرم افزار در پلت فرم موبایل (iOS، اندروید، ویندوز 7) با سازندگان مختلف سخت افزار (اپل، HTC، سامسونگ و گوگل) روش‌های تحویل متفاوت

(اپلیکشن بومی، اپلیکیشن وب موبایل) و پلت فرم‌های رایانشی متفاوت (تلفن‌های هوشمند، تبلت) دارند، (3) طراحی اپلیکیشن‌های موبایل آگاه از زمینه (آگاه از زمان، محل، و دستگاه) و (4) عدم قطعیت و چابکی توازن در مشخص کردن و تحلیل پیش نیازهای اپلیکیشن موبایل، هستند.

Williamson و همکاران [31] سه چالش متمایز را برای توسعه اپلیکیشن‌های موبایل لیست کردند، که عبارتند از (1) فرم فاکتور<sup>6</sup> و تکنولوژی ورودی کاربر که به صورت قابل توجهی در مقایسه با فرم نرم افزار قبلی متفاوت است، (2) قابلیت استفاده و طراحی تعامل کاربر برای اپلیکیشن موبایل نیاز به سرمایه گذاری بیشتر بر آن را تحریک می‌کند، به خصوص برای طراحی کاربر محور، و (3) انتخاب رویکرد اجرا برای اپلیکیشن موبایل (بومی، وب و هیبریدی)

El-Kassas و همکاران [7] چالش‌ها و محدودیت‌های توسعه اپلیکیشن موبایل را شناسایی کردند که عبارتند از:

• منابع محدود دستگاه موبایل مانند قدرت محاسباتی محلی، فضای ذخیره محدود، ارتباط برای دستگاه‌های موبایل که به شدت تحت تاثیر حرکت هستند [16].

• ناهمگنی OS موبایل، که در آن توسعه یک اپلیکیشن توسط iOS SDK تنها می‌تواند بر دستگاه‌های iOS اجرا شود، و با Android Apps که تنها می‌تواند بر دستگاه‌های اندروید اجرا شوند، مشابه است [24].

• ناهمگنی دستگاه‌ها ممکن است نیازمند نسخه‌های گوناگون همان اپلیکیشن باشد [24]، که در زمان توسعه اپلیکیشن موبایل قابلیت‌های محاسباتی و ساختارهای سخت افزاری متفاوت دستگاه‌ها باید در نظر گرفته شود (مانند اندازه صفحه نمایش، متدهای ورودی: صفحه لمسی، صفحه کلید، و کنترل از راه دور)

• تجربه کاربر، با تعریف یک واسط کاربر دوستانه و ساده برای اپلیکیشن‌های موبایل [22].

• پشتیبانی و حفظ اپلیکیشن، چرا که به روزرسانی مکرر پلت فرم موبایل ممکن است بر برخی از اپلیکیشن‌ها تاثیر گذارد که این اپلیکیشن‌ها ممکن است در نسخه‌های جدید غیرقابل استفاده شوند؛ بنابراین، نگهداری اپلیکیشن نیاز است [22]. علاوه بر این، مدیریت ورژن چالش برانگیز است چرا که کاربران ممکن است اپلیکیشن را به آخرین ورژن

---

<sup>6</sup> Form factor

در زمانی که منتشر می‌شود به روزرسانی نکند [24]. نگهداری اپلیکیشن برای پلت فرم‌های متفاوت به معنی تکرار همان به روزرسانی در همه نسخه‌های پلت فرم‌های متفاوت توسط تیم توسعه است [21].

• توسعه Cross-platform که بر توسعه همان اپلیکیشن برای پلت فرم‌های متفاوت به دلیل تفاوت در فروشندگان پلت فرم، زبان برنامه نویسی و ابزارهای توسعه تاکید دارد.

[6] Dye and Scarfone بر چالش‌های امنیتی متفاوتی در نتیجه غنی سازی اپلیکیشن‌های نرم افزار موبایل و صحبت در مورد ریسک‌های محتمل تحمیل شده به این دستگاه‌ها با توجه به عدم حضور استانداردهای توسعه و بهترین شیوه‌ها بحث کردند.

[17] Kaur and Kaur بر چالش‌های گوناگون توسعه اپلیکیشن موبایل که در توسعه اپلیکیشن‌های سنتی دیده نشده است بحث کردند؛ (1) تقسیم بندی که به معنی داشتن دستگاه‌های متعدد با نسخه‌های OS گوناگون در بازار است، (2) تست دستگاه‌های متفاوت، که در آن اتخاذ دستگاه جدید در هر مرتبه شبیه سازی زمانی که انواع گسترده ای از تلفن‌های همراه وجود دارد امکان پذیر نیست. (3) زمان نسبت به بازار باید با داشتن چرخه انتشار سریع و کوتاه کاهش یابد (5) تطبیق نسخه‌های جدید تر. (5) چند وظیفه ای بودن که در آن نشان دادن اپلیکیشن‌های بسیار به صورت همزمان؛ به دلیل صفحه نمایش کوچک، و با اپلیکیشن‌های متفاوتی که در پیش زمینه اجرا می‌شود، و مقدار باتری قابل توجهی که هدر می‌رود، سخت است. (6) فرم فاکتور با داشتن فرم متمایز دستگاه‌های موبایل مانند تلفن‌ها و تبلت‌ها، که طراحی اپلیکیشن برای آن‌ها بسیار متفاوت است. (7) تنوع، که به معنی کیفیت‌های مختلف گوشی‌ها از نظر اندازه صفحه نمایش، OEM و اپراتورها است.

Flora و همکاران [9] مسائل و چالش‌های منحصری را که در طول توسعه اپلیکیشن موبایل با آن مواجه می‌شویم که به مسائل سخت افزاری و نرم افزاری مربوط هستند را ارائه دادند. ابتدا، چالش‌های مربوط به سخت افزار شامل موارد زیر است:

• قابلیت‌های Cross-platform: شرکت‌های موبایل نیازمند طراحی و توسعه اپلیکیشن‌هایی هستند که قادر به اجرا در دستگاه‌های متعدد باشند چرا که بیشتر کسب و کارهایی که تنها بر پلت فرم‌های واحدی تمرکز می‌کنند، ممکن

است دستاوردهای اپلیکیشن‌ها را کاهش دهند. از این رو، ارائه یک اپلیکیشن cross-platform ممکن است به گسترش اپلیکیشن به پلت فرم‌های گوناگون کمک کند، اما حفظ اپلیکیشن در سراسر این پلت فرم‌ها با منابع محدود، کاملاً چالش برانگیز است.

- پیچیدگی‌های متفاوت سخت افزاری: در مقایسه با کامپیوترهای رومیزی، سخت افزار موبایل بسیار با توجه به حافظه، سرعت، پردازش گرافیکی، دسترس پذیری قدرت، جریان‌های داده بلادرنگ و غیره محدود هستند. علاوه بر این، در توسعه اپلیکیشن موبایل، توسعه دهندگان موبایل برخی از قابلیت‌هایی را که مقدار زیادی حافظه را مصرف می‌کنند و سرعت پردازش سریعی را می‌خواهند غیرفعال می‌سازند، به همین دلیل، حفظ کیفیت نرم افزار چالش برانگیز است. دوم، چالش‌های مربوط به نرم افزار عبارتند از:

- منابع ناشناخته: دنبال کردن دستورالعمل‌ها برای هر دستگاه موبایل در زمان ایجاد اپلیکیشن موبایل ضروری است. مطالعات نشان داده اند که تقریباً 10٪ از شرکت کنندگان در مورد عدم حضور منابع آگاه برای کد نویسی اپلیکیشن برای دستگاه‌های موبایل متفاوت نگران هستند.

- پیش نیازهای غیرقطعی و ناکافی: زمانی که پیش نیازهای پروژه غیرقطعی و ناکافی هستند، این دو مسئله در نظر گرفته می‌شوند. به همین ترتیب در آغاز پروژه تلاش‌های مورد نیاز را ارزیابی می‌کنیم و در زمان کار نیازهای کلاینت را تحلیل می‌کنیم که ممکن است در افزایش بارکاری توسعه اپلیکیشن نتیجه دهد.

- بودجه و زمان بندی: اشتباه است اگر بگوییم که توسعه اپلیکیشن مستلزم تلاش و بودجه کمی است، در حقیقت این فرآیند به خودی خود یک فرآیند پیچیده است که محدودیت‌ها و مراحل زیادی دارد، به خصوص در زمانی که توسعه دهندگان موبایل با بودجه محدود و زمان بندی محدود برای انجام کار در مهلت مقرر کار می‌کنند، آن‌ها ممکن است بین ارائه یک اپلیکیشن با کیفیت ضعیف یا تحویل اپلیکیشن در مدت زمان تعیین شده، مجبور شوند که یکی را انتخاب کنند.

- تجربه کاربر: از آنجایی که حرکت و حسگرها بخش مهمی از روند طراحی هستند، این مسئله یک چالش برای طراحی اپلیکیشن موبایل است.

- واسط کاربری: بهترین استفاده از فضای محدود صفحه نمایش دستگاه‌های موبایل، یکی از وظایف چالش برانگیز توسعه دهندگان موبایل است، و این مسئله باعث شده است که طراحی واسط کاربری امروزه اهمیت بیشتری پیدا کند.
- تکنولوژی ورودی کاربر: شناخت روش‌هایی متفاوت از صفحه کلید برای جمع‌آوری و ارسال اطلاعات کاربر به دستگاه‌های موبایل چالش برانگیز است.
- فرم فاکتورها: برای سر و کار داشتن با اندازه صفحه نمایش کوچک و رزولوشن کم و در عین حال توسعه یک اپلیکیشن موبایل کاملاً چالش برانگیز است، فرم فاکتور به معنی این است که تنها مقدار کمی داده می‌تواند در زمانی که کاربر نیازمند شفافیت و سادگی اطلاعات است، نشان داده شود.
- ارائه اپلیکیشن‌های با کیفیت: بررسی رسمی و تست فشرده برای تضمین اینکه اپلیکیشن‌ها همه پیش‌نیازهای کاربر را برآورده می‌سازند و یک نرم‌افزار با کیفیت به مشتریان تحویل داده می‌شود نیاز است، ممکن است حفظ این سطح از کیفیت بالا کاملاً چالش برانگیز دیده شود.
- پیچیدگی تست: تست اپلیکیشن با توجه به انواع دستگاه‌ها و سیستم عامل‌های متفاوت به یک کار چالش برانگیز تبدیل شده است. سناریوهای تست باید بر شبیه‌سازها، و بر دستگاه‌های واقعی برای تست موفقیت یا شکست غیرمنتظره اپلیکیشن‌های موبایل با تشخیص هر مسئله مربوط به ارتباط شبکه یا سخت‌افزار موبایل انجام شوند.
- تحلیل کاربران هدف: مهم است که در واقع نیازهای کلاینت‌های بالقوه را قبل از توسعه یک اپلیکیشن جدید بدانیم و تحلیل کنیم، چرا که شکست در شناسایی و تحلیل آن‌ها ممکن است به توسعه اپلیکیشنی منجر شود که هیچ ارزشی برای بازار هدف با رقابت سخت ندارد.
- امنیت و حفظ حریم شخصی: مدیریت انواع دستگاه‌ها گوناگون و OS بسیار چالش برانگیز است، به خصوص در صنعت‌هایی که به شدت نظارت شده هستند، مانند امور مالی، سلامت و بهداشت و غیره. یکی از تعهدات توسعه دهندگان موبایل حفاظت و ایمنی سازی داده از مبدا به مقصد است، که نرم‌افزار، سرورها و انتقال اطلاعات اضافی را در بر می‌گیرد.

می‌توان به وضوح دید که این مسائل ضروری هستند. به همین ترتیب، این مسائل باید در طول فرآیندهای توسعه اولیه به منظور کاهش اثر تصمیمات ضعیف در نظر گرفته شوند. همچنین، برای هر مسئله ذکر شده، راه‌های شناخته شده‌ای وجود دارد. حتی اگر هر پلت فرم موبایل بهترین شیوه‌های انحصاری خود را داشته باشد، بسیاری از این بهترین شیوه‌های اپلیکیشن موبایل در همه پلت فرم‌ها استاندارد هستند [19].

## 6. بهترین روش توسعه اپلیکیشن موبایل

مهم است که در زمانی که شیوه‌هایی برای تشخیص منظم موفقیت متد توسعه اپلیکیشن موبایل تشخیص داده می‌شود "بهترین روش" را داشته باشیم. این بهترین روش‌ها شامل مجموعه‌ای از استانداردها و راهنماهایی است که گمانه زنی را حذف می‌کند و کیفیت و سازگاری را در همه جنبه‌های SDLC اپلیکیشن موبایل تضمین می‌کند. به هر حال، این شیوه‌ها برای هر چیزی "نوش دارو" در نظر گرفته نمی‌شود، در عوض آن‌ها می‌تواند مرحله اولیه خوبی باشند که نیاز به بازنگری بیشتر را برای اعضای تیم توسعه حذف می‌کند [30].

راهنماهای پلت فرم: چند راهنمای طراحی متمایز برای هر پلت فرم موبایل وجود دارد، که شامل مجموعه‌ای از توصیه‌ها و قوانین برای رسیدن به کیفیت مطلوبی است که می‌تواند برای تجربیات کار، واسط گرافیکی کاربر (GUI)، و بسیاری از امکانات دیگر، برای پلت فرم SDK باشد [29]. برای مثال، کنسرسیوم شبکه جهانی وب یک مجموعه از توصیه‌ها را برای اپلیکیشن‌های وب موبایل صادر کرد [20]. اپل راهنمای برنامه نویسی اپلیکیشن را برای iOS منتشر کرد [13]. راهنمای توسعه دهنده برای اندروید بهترین روش را با توجه به سازگاری برنامه، عملکرد، دستورالعمل UI و غیره بررسی کرده است [27]. در نتیجه، توسعه دهندگان باید راهنمای کمکی برای توسعه یک اپلیکیشن موبایل موفق را در بر بگیرند.

بخش بعدی لیستی از بهترین شیوه‌های توصیه شده برای توسعه موفقیت آمیز اپلیکیشن موبایل را از ادبیات موضوعی با توجه به SDLC هدف قرار می‌دهد.

## 6.1 برنامه ریزی

تعیین متدولوژی طراحی یک کلید برای توسعه اپلیکیشن‌های موبایل است، به خصوص در محیط‌های پلت فرمی که در آن تلاش‌های متعددی ممکن است به صورت همزمان صورت گیرد [9,18]. بنابراین، این برای طراحی یک طرح مناسب قبل از شروع برنامه نویسی اپلیکیشن توصیه می‌شود، که یک استراتژی کارآمد را برای جذب کلاینت‌های بالقوه در نظر می‌گیرد.

## 6.2 پیش نیازها

این مرحله به عنوان مهم ترین فازی در نظر گرفته می‌شود که در آن بحث بر طرح‌های کسب و کار بعد از جمع آوری، تحلیل و مستندسازی پیش نیازهای کلاینت صورت می‌گیرد. استراتژی برنامه ریزی باید از بهترین متدها برای تعامل کاربر، عملکرد و استفاده از منابع محدود تشکیل شده باشد، دائما باید بررسی پیش نیازها تکرار شوند [9].

## 6.3 طراحی و معماری

طراحی مناسب ترین معماری برای اپلیکیشن‌های موبایل باید توسط توسعه دهندگان در نظر گرفته شود [9,18]. برخی از مطالعات بهترین روش را توسعه یک اپلیکیشن لایه لایه در نظر گرفته اند، که در آن سازگاری عملکرد اپلیکیشن موبایل در همه پلت فرم‌ها تضمین شده است [9]. همچنین ایجاد یک مولفه پلت فرم قابل استفاده مجدد توصیه شده است که می‌تواند به تسریع زمان توسعه کمک کند [10,18].

## 6.4 تجربه کاربر

• راهنماهایی: تعریف و مشخص کردن فونت، رنگ ها، طرح ها، تصاویر استفاده شده و غیره، مبنایی برای UX است. این راهنماها باید رفتارهای قابل پیش بینی را تحت برخی از شرایط مانند (راه رفتن کاربر، تغییر شبکه، تایم اوت) نشان دهند [30].



• برندینگ اپلیکیشن: برندینگ منظم در داخل یک اپلیکیشن و از طریق یک پرتفوی سازمانی یک تشابه و پیوست با کلاینت را ایجاد می‌کند [30].

• ابزارهای Wireframe and Prototyping: استفاده از ابزارهای Wireframe and Prototyping در فازهای اولیه SDLC به کلاینت کمک می‌کند که اجرای اپلیکیشن را بصری سازد. علاوه بر این، تغییر طراحی اپلیکیشن در فازهای اولیه به صورت قابل توجهی در مقایسه با فازهای دیگر ساده تر و ارزانتر است [30].

## 6.5 توسعه

با توجه به اینکه کدام پلت فرم موبایل یک سازمان برای توسعه اپلیکیشن موبایل انتخاب می‌شود، یک مجموعه سراسری از بهترین روش‌های فرآیند توسعه از ادبیات موضوعی به شرح زیر شناخته شده است:

- سطح مناسب مستندات کد، قابلیت خوانایی کد را افزایش می‌دهد، و این مسئله با رشد تیم توسعه، مهم تر می‌شود.
- تیم توسعه باید دانش مناسب تری داشته باشد و دقیقاً در پلت فرم‌های موبایل هدف تست شود [18].
- توسعه تکراری اپلیکیشن موبایل با داشتن چرخه‌های تکراری کوتاه با تحویل مستمر، به منظور اتخاذ زودتر بازخورد کاربر در فرآیند توسعه، که می‌تواند فرآیند اولویت بندی و توسعه این تغییرات را مشخص کند، انجام می‌گیرد [30].
- بررسی کد هم‌تا برای تضمین اینکه توسعه دهندگان به دنبال استانداردهای کد گذاری شناخته شده و شناسایی نقص‌ها در کد هستند باید صورت گیرد [30].

## 6.6 تست

- شبیه سازی و تست "روی دستگاه": اپلیکیشن‌ها باید با استفاده از شبیه سازها برای سنجش قابلیت استفاده و عملکرد در مراحل اولیه تست شوند، و تست نهایی باید بر روی دستگاه‌های موبایل واقعی انجام شود [9, 18, 30]
- پوشش تست: اپلیکیشن‌های باید با مرورگرهای متعدد، دستگاه‌ها و OS‌های متعدد برای شناسایی هر عملکرد غیرمعمولی تست شوند [9, 18, 30].

- تست مستمر: تست باید به صورت مکرر در هر تکرار برای ارزیابی تعیین هر اشتباه در مراحل اولیه فرآیند انجام شود که برطرف کردن آن‌ها را کم هزینه تر می‌سازد [9,30].
- خودکارسازی تست: خودکارسازی تست با استفاده از اسکریپت‌هایی به منظور کاهش زمان، تلاش و هزینه این فرایندها صورت می‌گیرد [30].
- سناریوهای خاص موبایل: تست اپلیکیشن از نظر شرایطی مانند فقدان حافظه، مصرف برق، ارتباطات شبکه‌ای، وقفه‌هایی از اپلیکیشن‌های متفاوت و غیره باید انجام شود.
- تست “رام نشده”: اجرای تست‌ها تحت شرایط جهان واقعی تصویری از عملکرد اپلیکیشن بر دستگاه کاربر می‌دهند، مانند اپلیکیشن‌های متفاوتی که همزمان، با حافظه ناکافی اجرا می‌شوند [30].

## 6.7 توسعه

- چند جنبه وجود دارد که سازمان‌ها باید در توسعه اپلیکیشن‌های موبایل در نظر بگیرند:
- برنامه‌ها باید شامل چرخه انتشار به خوبی تعریف شده استقرار اپلیکیشن‌های موبایل و یک تشریح صریح از محیط هاستینگ باشند که نیازمند مواردی ( تست، توسعه، تولید) است.
- خودکارسازی فرآیند تنظیم و نصب و راه اندازی؛ در زمان صرفه جویی می‌کند و در نهایت در صرفه جویی قابل توجهی در هزینه بوجود می‌آید [30].
- همیشه تضمین تایید شده توسعه دهنده برای جلوگیری و اجتناب از تاخیر ممکن در زمان استقرار اپلیکیشن‌ها در فروشگاه‌های اپلیکیشن تجاری به روز است [30].
- از تاخیرها باید با تایید اینکه همه ویژگی‌های اپلیکیشن قبل از استقرار به روز شده اند ( نسخه، قابلیت سازگاری؛ امضا و غیره) اجتناب کرد.

## 6.8 نگهداری و پشتیبان

فاز نگهداری با بر طرف کردن مسائل گوناگونی سر و کار دارد که کاربران با توجه به قابلیت سازگاری یا محدودیت‌های نرم افزار و سخت افزار با آن مواجه می‌شوند که در طول فاز تست شناخته نشده است، و شامل ویژگی‌های جدید توسعه/انتشار و عملکردها است [12]. چند توصیه از ادبیات موضوعی با توجه به این فاز در زیر لیست شده است:

- تحلیل اپلیکیشن: عملکرد اپلیکیشن موبایل در بازار باید توسط ترکیب یک ویژگی تحلیلی، برای کمک به تمایز بین مسائل فنی فعلی، و نشان دادن اینکه چه ویژگی‌های دیگر توسط کاربران استفاده نمی‌شود نظارت شود، لذا آن‌ها می‌توانند حذف شوند یا به سادگی در دسترس قرار گیرند [9, 18].

- مکانیزم بازخورد صریح: توسعه دهندگان باید یک مکانیزم بازخورد را از طریق ایمیل، شبکه اجتماعی یا درون اپلیکیشن‌های واقعی، برای کمک به کاربران برای بیان نظرات خود با توجه به آنچه در مورد اپلیکیشن دوست دارند یا ندارند، و گزارش مسائل فنی توسعه دهند، و در نتیجه آن مسائل فنی را حداقل سازند و بهبودهایی را فراهم کنند و طرح آینده اپلیکیشن را به روز سازند.

## 6.9 جنبه‌های دیگر

- امنیت: دستگاه‌های موبایل آسیب پذیر هستند، و این دستگاه‌ها باید قابلیت‌های امنیتی را از پلت فرم موبایل و با استفاده از ابزارهای امنیتی به روز و مناسب برای حفاظت از داده حساس ترکیب کنند [9, 18]. این مسئله می‌تواند با استفاده از موارد مانند: دسترسی، کنترل با استفاده از احراز هویت، تامین امنیت خدمات وب / API ها، رمزگذاری داده؛ مدیریت شود.

- سیاست حفظ حریم شخصی: سیاست حفظ حریم شخصی باید شفاف، واضح و بدون ابهام باشد، و جمع اوری داده، اشتراک گذاری داده و شیوه‌های استفاده را پوشش دهد. توسعه دهندگان موبایل سیاست حفظ حریم شخصی را در مورد مولفه‌های که در پلت فرم‌های متفاوت استفاده می‌شوند، برای تسهیل قابلیت نگهداری ارائه می‌دهند [9]. به منظور ایجاد سیاست موثر، برقراری ارتباط موثر و باز، استفاده شفاف و زبان ساده، توصیه می‌شود، که برای کاربران

کنترل و انتخاب‌هایی را با توجه به پذیرش آن‌ها فراهم می‌کند، و از داده کاربر حفاظت می‌کند و قابلیت پاسخ‌گویی را تضمین می‌کند [9].

## 7. نتیجه‌گیری

یک هدف اصلی مطالعه کسب درک بهتر از چالش‌های اصلی که در طول توسعه یک اپلیکیشن موبایل و برای تشخیص و شناسایی بهترین روش‌هایی که برای موفقیت استراتژی توسعه اپلیکیشن موبایل با آن مواجه می‌شویم است. بعد از بررسی بهترین شیوه‌ها، این بیانگر این است که این مسائل و چالش‌ها باید در طول فرآیند توسعه اولیه به منظور کاهش اثر تصمیمات نامناسب، در نظر گرفته شوند. همچنین، برای هر یک از این مسائل ذکر شده، راه‌حل‌های شناخته شده ای وجود دارد، حتی اگر هر پلت فرم موبایل دارای بهترین روش انحصاری خود باشد، بسیاری از این بهترین شیوه‌های توسعه اپلیکیشن موبایل در همه پلت فرم‌ها استاندارد هستند. این بهترین شیوه‌ها شامل مجموعه جامعی از استانداردها و راهنماهایی هستند که گمانه زنی‌ها را حذف می‌کند و کیفیت و سازگاری را در همه جنبه‌های SDLC اپلیکیشن موبایل تضمین می‌کند. مطالعه بهترین روش‌های کلیدی توصیه شده برای توسعه موفقیت آمیز اپلیکیشن‌های موبایل در ادبیات موضوعی با توجه به SDLC لیست شده و بررسی شده‌اند. این بررسی افراد و شرکت‌ها را برای برنامه ریزی و بهبود استراتژی‌های موبایل و اپلیکیشن‌های خود ارزیابی می‌کند.

مسائل؛ چالش‌ها و بهترین شیوه‌های بخش شده در بخش 5 و 6، می‌توانند به عنوان موضوعات پژوهشی بالقوه در توسعه اپلیکیشن‌های موبایل در نظر گرفته شوند و می‌تواند حد و وسعت پیش نیازهای پژوهشی و صنعت موبایل را نشان دهند. برای مطالعات آینده، یک بررسی می‌تواند در میان توسعه دهندگان اپلیکیشن موبایل در عربستان سعودی برای تحلیل دقیق تر در مورد چالش‌های محلی و بهترین شیوه‌ها انجام شود. کارهای بیشتری در این عرصه برای اعتبارسنجی بیشتر بهترین روش‌های اصلی و اجرای مطالعات موردی مناسب برای شناخت روش‌های استفاده انجام می‌شود که توسط افراد و شرکت‌ها در مبنای جهانی استفاده می‌شود. یک مثال با اپلیکیشن بومی، وب و ترکیبی برای متوجه شدن اینکه آیا مجموعه واحدی از بهترین روش‌ها برای هر گروه از متدهای توسعه اپلیکیشن موبایل وجود دارد

یا نه انجام می‌شود. اگر چه تعداد گسترده ای از اپلیکیشن‌های موبایل و تعداد گسترده ای از مسائل پیچیده حل نشده اند که نیازمند کار بیشتر هستند، وجود دارند.

## 8. REFERENCES

- [1] Anwar, W. 2015. *Software Quality Characteristics Tested For Mobile Application Development : Literature Review and Empirical Survey*. Blekinge Institute of Technology.
- [2] Ashishdeep, A. et al. 2016. Software Process Models for Mobile Application Development: A Review. *Computer Science and Electronic Journal*. 7, 1 (2016), 150–153.
- [3] Corral, L. et al. 2015. Software assurance practices for mobile applications. *Computing*. 97, 10 (2015), 1001–1022.
- [4] Corral, L. et al. 2013. Software development processes for mobile systems: Is agile really taking over the business? *2013 1st International Workshop on the Engineering of Mobile-Enabled Systems (MOBS)* (May 2013), 19–24.
- [5] Dehlinger, J. and Dixon, J. 2011. Mobile application software engineering: Challenges and research directions. *Workshop on Mobile Software ....* (2011).
- [6] Dye, S.M. and Scarfone, K. 2014. A standard for developing secure mobile applications. *Computer Standards & Interfaces*. 36, 3 (Mar. 2014), 524–530.
- [7] El-Kassas, W.S. et al. 2015. Taxonomy of Cross-Platform Mobile Applications Development Approaches. *Ain Shams Engineering Journal*. (Oct. 2015).
- [8] Flora, H.K. et al. 2014. Adopting an Agile Approach for the Development of Mobile Applications. *International Journal of Computer Applications*. 94, 17 (2014), 43–50.
- [9] Flora, H.K. et al. 2014. An Investigation into Mobile Application Development Processes: Challenges and Best Practices. *International Journal of Modern Education and Computer Science*. 6, 6 (Jun. 2014), 1–9.
- [10] Flora, H.K. et al. 2014. An Investigation on the Characteristics of Mobile Applications: A Survey Study. *Information Technology and Computer Science*. 6, 11 (2014), 21–27.
- [11] Flora, H.K. and Chande, S. V. 2013. A Review and Analysis on Mobile Application Development Processes using Agile Methodologies. *International Journal of Research in Computer Scienc*. 3, 4 (Jul. 2013), 8–18.
- [12] Inukollu, V.N. et al. 2014. Factors Influencing Quality of Mobile Apps: Role of Mobile App Development Life Cycle. *International Journal of Software Engineering & Applications (IJSEA)*. 5, 5 (2014).
- [13] iPhone Application Programming Guide: <https://developer.apple.com/develop/>.
- [14] Kakkar, K. et al. 2013. Risk Analysis in Mobile Application Development. *Confluence 2013: The Next Generation Information Technology Summit (4th International Conference)* (2013), 429–434.
- [15] Kaleel, S. and Harishankar, S. 2013. *Applying agile methodology in mobile software engineering: Android application development and its challenges*.
- [16] Kao, Y.-W. et al. 2011. A Cross-Platform Runtime Environment for Mobile Widget-Based Application. *2011 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery* (Oct. 2011), 68–71.
- [17] Kaur, A. and Kaur, K. 2015. Suitability of Existing Software Development Life Cycle (SDLC) in Context of Mobile Application Development Life Cycle (MADLC). *International Journal of Computer Applications*. 116, 19 (2015), 1–6.
- [18] Mobile Application Development: Challenges and Best Practices: 2012. <https://www.accenture.com/sa-en/insight-mobile-application-development-challenges-best-practices>. Accessed: 2016-11-08.
- [19] Mobile Architecture Best Practices: Best Practices for Mobile Application Design and Development: 2015. <http://www.westmonroepartners.com/en/Insights/White-Papers/Mobile-Architecture-Best-Practices>.
- [20] Mobile Web Application Best Practices: 2010. <https://www.w3.org/TR/mwapp/>.
- [21] The Developer's Guide: <https://developer.android.com/guide/components/index.html>.
- [22] Perchat, J. et al. 2013. Component based Framework to Create Mobile Cross-platform Applications. *Procedia Computer Science*. 19, (2013), 1004–1011.
- [23] Popa, M. 2013. Considerations Regarding the Cross-Platform Mobile Application Development Process. *Economy Informatics*. 13, 1 (2013), 40–52.
- [24] Quan, B. et al. 2012. A Cross-platform Application Development Environment Supported by Cloud Service. *2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems* (Jun. 2012), 1421–1427.
- [25] de Souza, L.S. and de Aquino, G.S. 2014. Mobile Application Development: How to Estimate the Effort? *Computational Science and Its Applications – ICCSA 2014: 14th International Conference, Guimarães, Portugal, June 30 -- July 3, 2014, Proceedings, Part V*. B. Murgante et al., eds. Springer International Publishing, 63–72.
- [26] Williamson, L. 2012. *A mobile application development primer. A guide for enterprise teams working on mobile application projects*.
- [27] Vithani, T. and Kumar, A. 2014. Modeling the Mobile Application Development Lifecycle. *Proceedings of the International MultiConference of Engineers and Computer Scientists* (Hong Kong, 2014).
- [28] Wasserman, A.I. 2010. Software engineering issues for mobile application development. *Proceedings of the FSE/SDP workshop on Future of software engineering research - FoSER '10* (New York, New York, USA, Nov. 2010), 397.
- [29] Whitepaper: Mobile Application Development Best Practices: <http://www.propelics.com/resources/whitepaper-mobile-application-development-best-practices/>.
- [30] Williamson, L. 2012. *A mobile application development primer. A guide for enterprise teams working on mobile application projects*.