



دانشگاه آزاد اسلامی واحد پرند

پایان نامه کارشناسی

رشته مهندسی فناوری اطلاعات

طراحی و پیاده سازی سیستم موبایل بانک با بهره گیری از زبان

برنامه سازی Android

نام دانشجو: ساناز احمدزاده

استاد راهنما: جناب آقای مهندس علیایی

پاییز ۱۳۹۲

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

**تقدیم به :**

**پدر و مادر عزیزم**

آنان که علی‌رغم تحمل سختی‌ها و دشواری‌های فراوان مسیر پرپیچ و خم کسب دانش و معرفت را  
برایم هموار نموده و یآوری دلسوز و فداکار، پشتیبانی محکم و مطمئن برایم بوده‌اند.

## استاد گرامی جناب آقای مهندس علیایی

بسی شایسته است از زحمات شما که با حسن خلق، فروتنی و دلسوزی همواره راهنما و مشوق من بوده‌اید تقدیر و تشکر نمایم. راهنمایی‌های کار ساز و سازنده، تلاش و کوشش حضرتعالی در تعلیم و تربیت، انتقال معلومات و تجربیات ارزشمند و حقیقتاً قابل ستایش است.

## چکیده

در پروژه حاضر، طراحی و پیاده‌سازی برنامه‌ی موبایل بانک در بستر اندروید مدنظر می‌باشد. در مرحله نخست به بیان اهداف اصلی سیستم موبایل بانک و بانکداری الکترونیک پرداخته می‌شود. از سوی دیگر اطلاعات لازم برای شروع کار از طریق منابع معتبر گردآوری می‌شود. این اطلاعات به طور کلی شامل مشخصات یک سیستم موبایل بانک می‌باشند. تحلیل امکان سنجی پروژه را می‌توان به دو بخش امکان سنجی عملیاتی و امکان سنجی تکنولوژی تقسیم نمود. بخش اول شامل بررسی اهداف پروژه، فرصت‌ها و تهدیدهای برنامه می‌باشد و بخش دوم شامل معرفی تجهیزات و تکنولوژی‌های مورد نیاز برای پیاده‌سازی پروژه است. در مرحله بعد، ایجاد نمای سیستماتیک پروژه که شامل نمودارهای مربوط به مستندسازی سیستم می‌باشد، مدنظر قرار گرفته است: نمودارهایی نظیر موردکاربر<sup>۱</sup>، فعالیت<sup>۲</sup>، کلاس<sup>۳</sup>، توالی<sup>۴</sup> به منظور نمایش اجزای سیستم و بیان ارتباط آنها تولید خواهند شد. همچنین جداول پایگاه داده و نمودار ارتباط بین آنها طراحی می‌شوند، از سوی دیگر طراحی صفحات اندروید و مواردی که در هر صفحه باید قرار گیرند در این مرحله که بدنه اصلی کار نام دارد، انجام می‌شود. در نهایت در قسمت سرور برنامه لایه‌های دسترسی به داده، منطق تجاری و موجودیت‌ها با زبان برنامه‌نویسی جاوا پیاده‌سازی می‌شوند.

کلیدواژه: موبایل بانک، اندروید، سرور، زبان جاوا

- 
- 1- Use Case
  - 2- Activity
  - 3- Class
  - 4- Sequence

## فهرست مطالب

فصل اول	۱
قلمرو پروژه	۱
۱-۱ مقدمه	۲
۲-۱ مقدمه‌ای بر بانکداری الکترونیک و موبایل بانک	۳
۱-۲-۱ بانکداری الکترونیک	۳
۲-۲-۱ تجارت بر پایه موبایل	۵
فصل دوم	۷
شناخت، تحلیل و امکان‌سنجی پروژه	۷
۱-۲ مقدمه	۸
۲-۲ شناخت	۹
۱-۲-۲ تعریف پروژه	۹
۳-۲ امکان‌سنجی	۹
۱-۳-۲ امکان‌سنجی عملیاتی	۹
۲-۳-۲ فرصت و تهدید	۱۰
۳-۳-۲ امکان‌سنجی زمانبندی	۱۱
۴-۳-۲ امکان‌سنجی تکنولوژی	۱۲
۴-۲ تجزیه و تحلیل	۱۲
۱-۴-۲ آشنایی با متدولوژی RUP	۱۲
۲-۴-۲ آشنایی با نمودارهای UML	۱۶
۳-۴-۲ نمودارهای مربوط به پروژه	۱۸
فصل سوم	۳۴

۳۴	..... سند چشم انداز
۳۵	..... ۱-۳ مقدمه
۳۶	..... ۲-۳ تاریخچه بازرگانی (تجدید نظر)
۳۶	..... ۱-۲-۳ جدول زمانبندی
۳۶	..... ۲-۲-۳ معرفی پروژه
۳۷	..... ۳-۲-۳ محدوده
۳۷	..... ۴-۲-۳ تعاریف، اصطلاحات و اختصارات
۳۷	..... ۵-۲-۳ منابع و مراجع
۳۸	..... ۳-۳ موقعیت تجاری محصول
۳۸	..... ۱-۳-۳ فرصت تجاری
۳۹	..... ۲-۳-۳ مشکلات
۴۱	..... ۴-۳ توصیف مشتری و ذینفعان
۴۱	..... ۱-۴-۳ اطلاعات ذینفعان
۴۱	..... ۲-۴-۳ اطلاعات مشتری
۴۲	..... ۳-۴-۳ محیط مشتری و ذینفع
۴۲	..... ۵-۳ نیازمندیهای اصولی محصول
۴۲	..... ۶-۳ نیازمندیهای محیطی
۴۳	..... فصل چهارم
۴۳	..... طراحی
۴۴	..... ۱-۴ مقدمه‌ای بر زبان برنامه نویسی اندروید
۴۴	..... ۱-۱-۴ تاریخچه اندروید

۴۵	۲-۱-۴	کپی‌رایت و حق امتیاز
۴۶	۳-۱-۴	سهام بازار اندروید
۴۷	۲-۴	اندروید مارکت چیست؟
۴۸	۳-۴	زبان برنامه‌نویسی اندروید
۴۸	۴-۴	اجزای تشکیل دهنده سیستم‌عامل اندروید
۴۹	۵-۴	محیط‌های برنامه‌نویسی اندروید
۴۹	۱-۵-۴	محیط برنامه‌نویسی اکلپس
۴۹	۲-۵-۴	محیط برنامه‌نویسی اندروید استودیو
۵۰	۳-۵-۴	IntelliJ IDEA محیط برنامه‌نویسی
۵۴	۶-۴	معرفی فایل‌های موجود در برنامه‌های اندروید
۵۴	۱-۶-۴	معرفی فایل‌های جزئی
۶۰	۲-۶-۴	معرفی فایل ظاهری
۶۲	۳-۶-۴	معرفی پوشه‌های موجود در برنامه
۶۶		فصل پنجم
۶۶		پیاده‌سازی
۶۷	۱-۵	مقدمه
۶۷	۲-۵	پیاده‌سازی پایگاه داده
۶۷	۱-۲-۵	معرفی جداول موجود در پایگاه داده
۶۹	۲-۲-۵	ارتباط میان جداول پایگاه‌داده
۶۹	۳-۵	معرفی معماری برنامه
۶۹	۱-۳-۵	اجزای تشکیل دهنده معماری سه لایه



۷۱	۴-۵ نحوه‌ی برقراری ارتباط با پایگاه‌داده در جاوا
۷۳	۵-۵ نحوه ارتباط بین سرور و کاربر
۷۳	۱-۵-۵ مفهوم REST
۷۳	۲-۵-۵ مفهوم WebSocket
۷۳	۳-۵-۵ مفهوم Json
۷۴	۶-۵ معرفی فعالیت‌ها و صفحات موجود در برنامه
۷۴	۱-۶-۵ صفحه مربوط به ورود
۷۵	۲-۶-۵ صفحه مربوط به منو اصلی
۷۶	۳-۶-۵ صفحه مربوط به تنظیمات حساب
۸۰	۴-۶-۵ صفحه مربوط به انتقال وجه
۸۲	۵-۶-۵ صفحه نمایش تراکنشها
۸۳	۶-۶-۵ صفحه خروج
۸۴	۷-۶-۵ صفحه مربوط به تغییر رمز عبور
۸۵	فصل ششم
۸۵	جمع بندی و نتیجه‌گیری
۸۶	۱-۶ نتیجه‌گیری
۸۶	۲-۶ پیشنهادات
۸۹	منابع

## فهرست جداول

جدول ۱-۲	زمانبندی پروژه.....	۱۱
جدول ۲-۲	سناریو مورد کاربر ورود.....	۱۹
جدول ۳-۲	سناریو مربوط به مورد کاربر لغو عملیات.....	۲۰
جدول ۴-۲	سناریو مربوط به مورد کاربر عملیات بانکی.....	۲۱
جدول ۵-۲	سناریو مورد کاربر تنظیمات حساب.....	۲۲
جدول ۶-۲	سناریو مربوط به مورد کاربر اضافه کردن حساب.....	۲۳
جدول ۷-۲	سناریو مربوط به مورد کاربر نمایش مشخصات حساب.....	۲۴
جدول ۸-۲	سناریو مربوط به مورد کاربر انتقال وجه.....	۲۵
جدول ۱-۳	زمان بندی تهیه سند.....	۳۶
جدول ۲-۳	تعاریف موجود در سند.....	۳۷
جدول ۳-۳	مسئله عدم اطلاع رسانی.....	۳۹
جدول ۴-۳	مسئله شبکه و ارتباطات.....	۳۹
جدول ۵-۳	مسئله امنیت.....	۴۰
جدول ۶-۳	مسئله تفاوت در نسخه‌های اندروید.....	۴۰
جدول ۷-۳	اطلاعات ذینفعان.....	۴۱
جدول ۸-۳	اطلاعات مشتری.....	۴۱
جدول ۹-۳	جدول اطلاعات مربوط به محیط و مشتری.....	۴۲
جدول ۱-۵	جدول کاربران.....	۶۷
جدول ۲-۵	جدول حسابها.....	۶۸
جدول ۳-۵	جدول تراکنشها.....	۶۸

## فهرست اشکال

- شکل ۲-۱ امتداد RUP..... ۱۴
- شکل ۲-۲ نمایش نمودار مورد کاربرد تولید شده در نرمافزار رشنال رز..... ۱۸
- شکل ۲-۳ نمایش نمودار توالی ورود..... ۲۷
- شکل ۲-۴ نمایش نمودار توالی مشاهده حساب..... ۲۸
- شکل ۲-۵ نمایش نمودار توالی مشاهده تراکنشها..... ۳۰
- شکل ۲-۶ نمایش نمودار فعالیت..... ۳۱
- شکل ۲-۷ نمودار کلاس سرور..... ۳۲
- شکل ۲-۸ نمودار کلاس کاربر..... ۳۳
- شکل ۴-۱ نمایش محل کپی کردن فایل JDK..... ۵۰
- شکل ۴-۲ نمایش مرحله اول تنظیمات متغیرهای سیستمی..... ۵۱
- شکل ۴-۳ نمایش مرحله دوم تنظیمات متغیرهای سیستمی..... ۵۱
- شکل ۴-۴ نمایش سوم تنظیمات متغیرهای سیستمی..... ۵۲
- شکل ۴-۵ نمایش مرحله چهارم تنظیمات متغیرهای سیستمی..... ۵۲
- شکل ۴-۶ نمایش مرحله پنجم تنظیمات متغیرهای سیستمی..... ۵۳
- شکل ۴-۷ نمایش مرحله ششم تنظیمات متغیرهای سیستمی..... ۵۳
- شکل ۴-۸ شکل مربوط به نمایش اطلاعات مربوط به کیت توسعه جاوا..... ۵۴
- شکل ۴-۹ کدهای مربوط به کلاس فعالیت..... ۵۶
- شکل ۴-۱۰ کدهای مربوط به فایل ظاهری..... ۶۲
- شکل ۴-۱۱ نمایش پوشه حاصله..... ۶۳
- شکل ۵-۱ نمایش ارتباط میان جداول پایگاه داده..... ۶۹
- شکل ۵-۳ صفحه ورود..... ۷۴
- شکل ۵-۴ صفحه منو اصلی..... ۷۵
- شکل ۵-۵ صفحه تنظیمات حساب..... ۷۶
- شکل ۵-۶ صفحه اضافه کردن حساب جدید..... ۷۷

- شکل ۵-۷ صفحه حذف حساب و ارسال پیام ..... ۷۸
- شکل ۵-۸ صفحه تأیید حذف حساب ..... ۷۹
- شکل ۵-۹ صفحه ارسال پیام ..... ۷۹
- شکل ۵-۱۰ صفحه انتقال وجه ..... ۸۰
- شکل ۵-۱۱ صفحه گرفتن تأیید ..... ۸۱
- شکل ۵-۱۲ صفحه نمایش تراکنشها ..... ۸۲
- شکل ۵-۱۳ صفحه خروج از برنامه ..... ۸۳
- شکل ۵-۱۴ صفحه تغییر رمز عبور ..... ۸۴

## فصل اول

### قلمرو پروژه

## ۱-۱ مقدمه

رشد و گسترش روزافزون فناوری ارتباطات، انقلابی را در ابعاد مختلف زندگی انسانها و عملکرد سازمانها ایجاد کرده است. این فناوری روشهای کارکرد و نگرش افراد، سازمانها و دولتها را دگرگون ساخته و باعث ایجاد صنایع نوین، مشاغل جدید و خلاقیت در انجام امور شده است. ظهور پدیده‌هایی چون کسب و کار الکترونیک<sup>۱</sup>، تجارت الکترونیک<sup>۲</sup> و بانکداری الکترونیک<sup>۳</sup> از نتایج عمده نفوذ و گسترش فناوری اطلاعات در بعد اقتصادی است. کشور ما در عرصه حضور و به‌کارگیری تجارت و بانکداری الکترونیک، کشوری جوان است و تا رسیدن به وضعیت مطلوب راه درازی در پیش رو دارد. در زمینه بانکداری الکترونیک تاکنون فعالیتهای گسترده‌ای صورت گرفته، لیکن نبود برخی زیر ساختها در ابعاد مختلف موجب کندی این فعالیتها شده است. روند رو به رشد کاربرد انواع گوشی‌های هوشمند<sup>۴</sup> و رایانه‌های لوحی<sup>۵</sup> و نیز استقبال گسترده از خدمات ارزش افزوده آنها موجب گردیده تا درخواستهای متنوعی از خدمات بانکی بر روی موبایل، به عنوان ابزاری مهم در بانکداری الکترونیک مطرح گردد، لذا به منظور ایجاد وحدت رویه در ارائه خدمات بانکداری الکترونیک و توسعه و تکمیل خدمات آن، طراحی و تهیه سامانه جدید موبایل بانک<sup>۶</sup> انجام شد. سیستم همراه بانک مشتریان را قادر می‌سازد با استفاده از تلفن همراه خود، بدون حضور در بانک و در زمان دلخواه، اقدام به امور بانکی نمایند. همچنین، ضمن سرعت و دقت بالا در ارائه پاسخ و پاسخگویی شبانه روزی، میتواند باعث صرفه جویی در انرژی گردد.

در این پروژه سعی بر ارائه مفاهیم، ویژگیها، ابزارهای بانکداری موبایل و پیاده سازی برنامه موبایل بانک<sup>۶</sup> تحت سیستم عامل اندروید شده است.

- 
- 1- Electronic Business
  - 2- Electronic Commerce
  - 3- Electronic Banking
  - 4- Smartphone
  - 5- Tablet
  - 6- Mobile Banking

## ۲-۱ مقدمه‌ای بر بانکداری الکترونیک و موبایل بانک

### ۱-۲-۱ بانکداری الکترونیک

برای شناخت هر پدیده‌ای لازم است تا ابتدا تعریف مشخصی از آن پدیده و عوامل و متغیرهای مرتبط با آن ارائه کرد. برای بانکداری الکترونیک تعاریف گوناگونی ارائه شده است که می‌توان به تعاریف زیر اشاره کرد.

- فراهم آوردن امکان دسترسی مشتریان به خدمات بانکی با استفاده از واسطه‌های ایمن و بدون حضور فیزیکی
- استفاده مشتریان از اینترنت برای سازماندهی، آزمایش و یا انجام تغییرات در حسابهای بانکی خود و بانکها برای ارائه عملیات و سرویسهای بانکی
- ارائه مستقیم خدمات و عملیات بانکی جدید و سنتی به مشتریان از طریق کانالهای ارتباطی الکترونیکی.

### ۱-۲-۱-۱ سطوح بانکداری الکترونیک

بانکداری الکترونیک شامل سیستم‌هایی است که مشتریان موسسات مالی را قادر می‌سازد تا در سه سطح اطلاع‌رسانی، ارتباط و تراکنش از خدمات و سرویس‌های بانکی استفاده کنند.

- اطلاع‌رسانی: این سطح ابتدایی‌ترین سطح بانکداری اینترنتی است. بانک، اطلاعات مربوط به خدمات و عملیات بانکی خود را از طریق شبکه‌های عمومی یا خصوصی معرفی می‌کند.

- ارتباطات: این سطح از بانکداری اینترنتی امکان انجام مبادلات بین سیستم بانکی و مشتری را

فراهم می‌آورد. ریسک این سطح در بانکداری الکترونیک بیشتر از شیوه سنتی است و بنابراین، برای جلوگیری و آگاه ساختن مدیریت بانک از هرگونه تلاش غیر مجاز برای دسترسی به شبکه اینترنتی بانک و سیستم‌های رایانه‌ای به کنترل‌های مناسبی نیاز است.

تراکنش: این سیستم متناسب با نوع اطلاعات و ارتباطات خود، از بالاترین سطح ریسک برخوردار است و باید سیستم امنیتی قوی بر آن حاکم باشد. در این سطح مشتری در یک ارتباط متقابل قادر است تا عملیاتی چون پرداخت صورتحساب، صدور چک، انتقال وجه و افتتاح حساب را انجام دهد.

#### ۱-۲-۱-۲ کانال‌های بانکداری الکترونیک

برای ارائه خدمات بانکداری الکترونیک کانال‌های متعددی وجود دارد که برخی از آنان عبارتند از: رایانه‌های شخصی، تلفن ثابت و همراه، و ماشین‌های خودپرداز و.... در روش رایانه‌های شخصی، بانک از طریق ایجاد یک پایگاه اینترنتی و معرفی آن به مشتریان، با آنها ارتباط متقابل برقرار کرده و ارائه خدمت می‌کند. در روش بانکداری تلفنی، تلفن (اعم از تلفن ثابت و همراه) وسیله ارتباطی بانک با مشتریان خود بوده و از این طریق خدمات بانکی عرضه می‌شود. با استفاده از ماشین‌های خودپرداز نیز بانکها می‌توانند خدمات متنوعی از قبیل برداشت نقدی، سپرده‌گذاری، انتقال وجوه، پرداخت صورتحساب و غیره را به مشتریان خود ارائه دهند. [۱]



## ۱-۲-۲-۲-۱ تجارت بر پایه موبایل

تجارت بر پایه موبایل<sup>۱</sup> در واقع توانایی انجام تجارت با استفاده از هر گونه تجهیزات همراه الکترونیکی شامل تبلت، تلفن‌های هوشمند و... می‌باشد، هرگونه معامله شامل انتقال مالکیت یا حق استفاده از کالا و خدمات با استفاده از فناوری موبایل را تجارت بر پایه موبایل گویند. موبایل بانک یا بانکداری موبایل، هم در حوزه تجارت بر پایه موبایل و هم در حوزه بانکداری الکترونیک قرار می‌گیرد.

## ۱-۲-۲-۱-۱ تکنولوژی‌های ارتباطی در موبایل

اساساً در استاندارد سیستم جهانی برای ارتباطات موبایل<sup>۲</sup>، دریافت و انتقال اطلاعات به سه روش صورت می‌گیرد:

خدمات تکمیلی بدون ساختار داده‌ها (USSD)<sup>۳</sup>:

یک تکنولوژی منحصر به فرد در استاندارد جهانی ارتباطات موبایل است. قابلیت است برای حمایت از انتقال اطلاعات در کانال‌های سیگنالی شبکه‌ها در واقع تکنولوژی جلسه گرا (تراکنش گرا) می‌باشد و تراکنش‌ها فقط در طی جلسات اتفاق می‌افتند (همزمان ارسال و دریافت صورت می‌گیرد)

سرویس پیام کوتاه (SMS)<sup>۴</sup>:

این یک سرویس پیام متنی است که قادر به ارسال پیام کوتاه (۱۶۰-۱۴۰) کاراکتری می‌باشد، ذخیره و ارسال پیام توسط یک مرکز سرویس دهی صورت می‌گیرد.

---

1- Mobile commerce

2- GSM (Global System for Mobile Communications)

3- Unstructured Supplementary Service Data

4- Short Message Service

بسته سرویس رادیویی (GPRS)<sup>۱</sup>:

GPRS یا نسل ۲/۵ تلفن‌های همراه، یک سرویس رادیویی بی‌سیم، جهت ارسال دیتا است که امکان ارسال دیتا را با سرعت ۱۷۱/۲ کیلو بیت بر ثانیه، امکان پذیر می‌سازد. با به کارگیری این تکنولوژی در شبکه تلفن همراه، امکاناتی نظیر مرور در اینترنت، ارتباطات تصویری، پیام‌های چند رسانه‌ای فراهم می‌شود این سرویس به شبکه‌ها این امکان را می‌دهد که خدمات اینترنتی پر ظرفیت را به صورت دائمی و بدون وقفه، به مشترکان خود ارائه دهند.[۲]

---

1- General packet radio service

## فصل دوم

### شناخت، تحلیل و امکان‌سنجی پروژه

## ۲-۱ مقدمه

در دهه‌های اخیر با توجه به تحولات گسترده و تأثیرگذار بازارهای رقابتی در صنایع و بنگاه‌های مختلف به ویژه بانک‌ها، مدیریت و برنامه‌ریزی استراتژیک به مفهومی توأم با تفکر استراتژیک و مبتنی بر بازار و بازاریابی، به عنوان مقوله‌ای اساسی و فراگیر در همه‌ی مراحل و سطوح برنامه‌ریزی استراتژیک تبدیل شده‌است. بنابراین با توجه به شرایط حاکم بر اقتصاد جهانی و بازار رقابت بانک‌ها، برنامه‌ریزی استراتژیک مناسب در بانک‌ها که بتواند موفقیت این سازمان‌ها را در بازار رقابتی تضمین کند، باید مبتنی بر بازار باشد و روح بازاریابی در کالبد آن دمیده شود. در یک تعریف جامع می‌توان بازاریابی بانکی را فرآیندی دانست که در نتیجه‌ی آن بازار و مشتریان و نیازهای آن‌ها، توانمندی‌ها و خدمات بانک و نیز عملکرد رقبا در سطوح مختلف شناسایی شده و خدمات بانکی با شرایط مدنظر مشتریان هدف به شکلی بهتر از رقبا به آنان ارائه می‌شود، به گونه‌ای که رضایت مشتری فراهم شود و رابطه‌ای بلندمدت بین بانک و مشتری به صورت سودده برای هر دو طرف ایجاد شود و تداوم یابد. این رابطه باید تا جایی عمق یابد که مشتری، مشتریان جدیدی را به بانک معرفی کند. بنا به این تعریف، در شرایط کنونی، بازاریابی نمی‌تواند به یک بخش، اداره یا دایره از بانک محدود شود. بلکه همه‌ی بخش‌ها، ارکان، کارکنان و مدیران و نیز همه‌ی طرح‌ها، برنامه‌ها و اقدامات بانک باید با روحیه‌ی بازارمحور، به اهمیت بازاریابی توجه کنند. امروزه مشتریان بانکی علاوه بر سوددهی بانک به سرعت، دقت و امنیت موجود در خدمات هم توجه می‌کنند. و در زمینه موبایل بانک چون پدیده‌ای نوظهور می‌باشد، بانک‌ها در یک محیط رقابتی سعی بر جذب بیشتر مشتریان و ایجاد رضایت در آنها می‌باشند. در این فصل سعی بر شناخت نیازها، بررسی اهداف پروژه و امکان‌سنجی آنها شده‌است.

## ۲-۲ شناخت

### ۱-۲-۲ تعریف پروژه

استفاده از خدمات نوین بانکی با توجه به اهمیت مواردی نظیر مدیریت زمان، کاهش ترافیک و تسریع در انجام عملیات بانکی، پرداخت انواع قبوض و .... تبدیل به یکی از الزامات زندگی امروزه شده است. در این راستا موبایل بانک، طیف وسیعی از خدمات بانکی را در هر زمان و هر مکان در اختیار مشتریان قرار می-دهد. این خدمات به مشتری امکان می-دهد از طریق نصب برنامه متناسب با سیستم عامل اندروید و جاوا بر روی تلفن همراه خود بتواند پاره‌ای از امور بانکی خود را انجام دهد. ما در این پروژه پیشنهادی تکنیک-های لازم برای پیاده‌سازی موبایل بانک، تحت سیستم عامل اندروید را معرفی خواهیم کرد.

## ۲-۳ امکان سنجی

### ۱-۳-۲ امکان سنجی عملیاتی

#### ۱-۱-۳-۲ اهداف پروژه

- مشاهده موجودی حساب‌ها
- مشاهده گردش حساب‌ها
- انتقال وجه
- تنظیمات (تغییر رمز عبور، تعریف شماره حساب جدید یا حذف آن)

#### ۲-۱-۳-۲ مراحل انجام پروژه

(۱) برنامه‌ریزی: زمانبندی و مستند سازی نیازمندی‌های سیستم

۲) شناخت و تحلیل: شناخت روابط میان اجزا، تحلیل نیازمندی‌های نرم‌افزار و درک ماهیت برنامه، روال‌های مورد نیاز، رفتار و کارایی برنامه، تعیین دامنه اطلاعات عملیات مورد نظر شامل (ورودی، خروجی، پردازش‌ها) و مستندسازی نیازها و تولید نمودارهای مربوطه

۳) طراحی: طراحی عناصر تحلیل شده مورد نیاز در ساختمان داده، رابطه‌ها و موجودیت‌ها، مدل سازی آنها، انتخاب بسترها، تکنولوژی‌ها و روش‌های پیاده سازی.

۴) پیاده سازی

۵) تست: اجرای آزمایشی برنامه

## ۲-۳-۲ فرصت و تهدید

مزایای بانکداری موبایل را می‌توان از دوجنبه مشتریان و موسسات مالی مورد توجه قرار داد. از دید مشتریان می‌توان به صرفه‌جویی در هزینه‌ها، صرفه‌جویی در زمان و دسترسی به کانالهای متعدد برای انجام عملیات بانکی نام برد. از دید موسسات مالی می‌توان به ویژگیهایی چون ایجاد و افزایش شهرت بانکها در ارائه نوآوری، حفظ مشتریان علی‌رغم تغییرات مکانی بانکها، ایجاد فرصت برای جست‌جوی مشتریان جدید در بازارهای هدف، گسترش محدوده جغرافیایی فعالیت و برقراری شرایط رقابت کامل را نام برد. در هر حال ایجاد و توسعه بانکداری الکترونیک مستلزم برخورداری از برخی زیرساختهای مناسب اقتصادی و اجتماعی است. اهم این زیرساخت‌ها عبارتند از: شبکه‌های ارتباطی و مخابراتی مناسب، امنیت تبادل اطلاعات، زیرساختهای حقوقی و قانونی مناسب، آمادگی فرهنگی جامعه و بنگاه‌های اقتصادی برای پذیرش و استفاده از خدمات بانکداری الکترونیک.

بنابراین برای توسعه و گسترش مناسب بانکداری الکترونیک و تجارت الکترونیک در کشور باید نسبت به ایجاد زیرساخت‌های مناسب مخابراتی، امنیت در تبادل اطلاعات، تدوین قوانین و مقررات حقوقی مناسب، بسترسازی فرهنگی و آشنا نمودن مردم و بنگاه‌های اقتصادی با مزایای تجارت الکترونیکی و پرداخت الکترونیک، تسریع در شکل‌گیری دولت الکترونیک، به طور جدی اهتمام ورزید.

مهمترین مساله برای کاربرانی که از خدمات بانکداری موبایل استفاده می‌کنند، امنیت است و اگر این مساله مورد توجه قرار نگیرد، بانکداری موبایل با استقبال مواجه نخواهد شد. به نظر بعضی از کارشناسان، بانکداری موبایل به طور بالقوه بسیار امن تر از بانکداری از طریق اینترنت است.

### ۳-۳-۲ امکان سنجی زمانبندی

#### ۱-۳-۳-۲ جدول زمانبندی پروژه

جدول ۱-۲ زمانبندی پروژه

شماره ردیف	مراحل انجام پروژه	زمان مورد نیاز(روز)
۱	برنامه ریزی اجرای برنامه	روز ۱۵
۲	شناخت و تحلیل نیازمندی‌ها : تهیه سند چشم‌انداز طراحی نمودارها	مجموعاً ۲۵ روز ۱۰ روز ۱۵ روز
	ایجاد پایگاه داده	روز ۱۵
۳	قالب گرافیکی برنامه طراحی	روز ۲۰
۴	پیاده سازی	روز ۳۰
۵	تست سیستم	روز ۷

## ۲-۳-۴ امکان سنجی تکنولوژی

### ۲-۳-۴-۱ روش‌ها و تکنولوژی‌های مورد استفاده در پیاده سازی

- انتقال و دریافت اطلاعات از اینترنت به واسطه‌ی وب سوکت<sup>۱</sup>
- ارتباط با بانک اطلاعاتی از طریق JDBC<sup>۲</sup>
- نقل و انتقال اطلاعات بر مبنای JSON<sup>۳</sup>
- راه اندازی سرور بر اساس Rest<sup>۴</sup>
- معماری MVC<sup>۵</sup>

در فصل طراحی از موارد نام برده، به بیان جزئیات بیشتری می‌پردازیم.

## ۲-۴ تجزیه و تحلیل

### ۲-۴-۱ آشنایی با متدولوژی RUP<sup>۶</sup>

پیش از هرچیز بهتر است ابتدا تعریف متدولوژی را بررسی کنیم.

### ۲-۴-۱-۱ تعریف متدولوژی

مجموعه‌ای از روشها، فنون و ابزارهای تحلیل و طراحی سیستم است که در چهارچوب یک انگاره مدل‌سازی مبتنی بر یک الگوی مفهومی برای ساماندهی روند توسعه سیستم‌ها به روشی نظام‌مند به کار بسته می‌شود.

---

1- Web Socket  
2- Java Data Base Connectivity  
3- JavaScript Object Notation  
4- Representational State Transfer  
5- Model-View-Controller  
6- Rational Unified Process



## ۲-۱-۴-۲ تعریف RUP

در فرآیند انجام یک پروژه، مهندسان نرم‌افزار به این نتیجه رسیده بودند که برای رسیدن به هدفی موفقیت‌آمیز، نیاز دارند تا اطلاعاتی را ثبت و بررسی کنند و بر اساس یک روش قانونمند تولید و یا بهبود یک نرم‌افزار را به ثمر برسانند. لذا در جهت تلاش‌های ایشان، روشی سودمند توسط شرکت رشنال، خلق شد و به همگان ارائه گردید. نام این متودولوژی را RUP نهادند. از آنجا که RUP میان متودولوژی‌های مختلفی که جهت تحلیل، طراحی و پیاده‌سازی سیستم برای تولید نرم‌افزار به کار گرفته می‌شد سودمندتر بود و از طرفی دیگر، طبق آمار به گزارش رویترز از سال ۲۰۰۱ تا کنون یکی از پرکاربردترین متودولوژی مورد استفاده در شرکت‌های نرم‌افزاری می‌باشد. به همین علت امروزه به نوعی یک متودولوژی استاندارد برای تولید و ارتقاء نرم‌افزار در نظر گرفته می‌شود.

از اهداف این متودولوژی در اصل ارائه یک نرم‌افزار (محصول) با کیفیت بالا می‌باشد. این روش بر اساس یک برنامه از قبل ریخته شده عمل می‌کند و در مواقع لزوم برخی از بخش‌های آن تکرار می‌گردند، از طرفی دیگر زمان مورد نیاز و بودجه مورد نیاز را تخمین می‌زند تا بتواند نظر سفارش‌دهنده را جلب کند. در این تکنیک از زبان مدل‌سازی UML<sup>۱</sup> که زبانی جهانی برای مهندسی نرم‌افزار می‌باشد، استفاده می‌گردد. دلیل اصلی پرکاربرد بودن متودولوژی RUP، انعطاف‌پذیری آن است و در مراحل مختلف بنا به نیاز محصول و سفارش‌دهنده می‌توان برخی از فازها را مجدداً تکرار کرده و بیشتر روی آن‌ها تمرکز کنیم، در واقع همانند یک چهارچوب بر کیفیت و موفقیت محصول کنترل داشته و فازها<sup>۲</sup> و دیسپلین<sup>۳</sup> آنها به نحوی ترتیب‌بندی شده‌است که احتمال ریسک را کاهش می‌دهد.

---

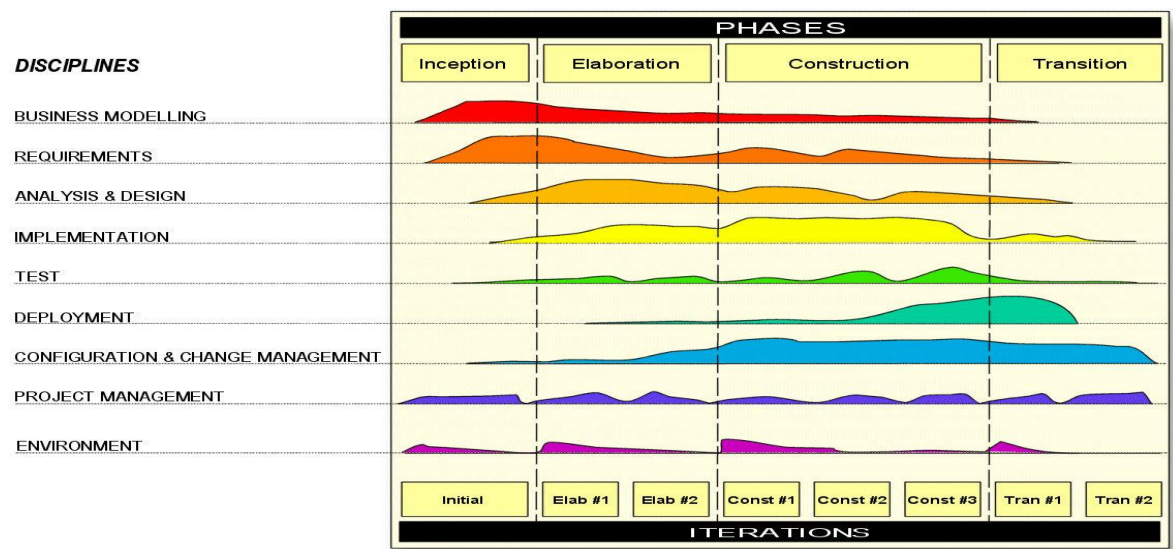
1- Unified Modeling Language

2- phases

3- disciplines

## ۲-۴-۱-۳ فازهای RUP

متدولوژی مورد بررسی دارای قسمت‌های مختلفی است، ابتدا دارای چهار فاز اصلی می‌باشد که به ترتیب اجرا می‌شوند و بنا به نیاز روی هر فاز مکث و یا تکرار صورت می‌گیرد تا بهترین نتیجه حاصل گردد. در حین انجام فازها تعدادی دیسپلین در پروژه انجام می‌شود که در شکل زیر در سمت چپ تصویر به صورت عمودی قرار گرفته‌است. شکل ۱-۲ ارتباط و ترتیب فازها و دیسپلین‌ها را نمایش می‌دهد.



شکل ۱-۰-۱ متد RUP

## ۲-۴-۱-۴ معرفی اهداف فازهای RUP

الف: فاز آغازین<sup>۱</sup>: در این فاز محدوده پروژه و سناریوی اصلی تعیین می‌شود و زمان لازم و هزینه لازم تخمین زده می‌شود. سپس شرایط را سنجیده و محیط مورد نیاز جهت اجرای پروژه تعیین می‌گردد.

ب: فاز تعیین<sup>۲</sup>: در این مرحله اطمینان از معماری تعیین شده حاصل می‌گردد و بررسی می‌شود که آیا پیشنهادات مرحله قبل صحت دارند یا خیر. در صورت لزوم، بسته به محصول، محیطی جهت پشتیبانی ایجاد می‌گردد، خواسته‌ها و تصمیمات جدید از جانب کاربر اخذ می‌شود و یک نمونه اولیه ایجاد می‌گردد.

1- Inception  
2- Elaboration

ضمناً این موضوع که آیا مراحل پروژه طبق زمان و هزینه محاسبه شده پیش می‌رود نیز کنترل می‌گردد، در غیر این صورت هزینه و زمان جدید جایگزین می‌شود. این مرحله در پروژه‌ها بسیار حیاتی بوده و روی آن مکث می‌گردد.

پ: فاز ساخت<sup>۱</sup>: در این مرحله رعایت این نکته بسیار اهمیت دارد و آن این است که از دوباره کاری جلوگیری شود، نسخه‌هایی از پروژه در این مرحله آماده می‌گردد تحلیل، طراحی، پیاده‌سازی و تست در این مرحله صورت می‌گردد. در این مرحله محصولی کامل با روش تکرار ساخته می‌شود تا نمونه بهبود یافته‌ی آن استخراج گردد.

ت: فاز انتقال<sup>۲</sup>: در این مرحله همان‌طور که از نامش پیداست بیشتر بر روی آموزش کاربران و مسئولان تمرکز می‌گردد تا آنها را با مسائل اساسی در مورد محصول آشنا کنند و عیب‌ها برطرف شده و موارد اضافی در صورت وجود در محصول درج گردند. مهم‌ترین نکته در این فاز بررسی آن است که آیا آنچه ساخته شده با آنچه که در چشم‌انداز مستند شده هم‌خوانی دارد یا خیر.

## ۲-۴-۱-۵ دیسپلین‌های مورد استفاده در تولید یا بهبود نرم‌افزار

با توجه به شکل ۱-۲ و خط عمودی نمودار، در زیر به نام بردن دیسپلین‌ها می‌پردازیم:

۱. مدل سازی کسب و کار.

۲. تعیین نیازمندی‌ها.

۳. تحلیل و طراحی.

۴. پیاده‌سازی.

۵. آزمون (تست).

۶. استقرار.

موارد فوق دیسپلین‌های اصلی نامیده می‌شوند، اما دیسپلین‌های فرعی عبارتند از:

۱. آرایش پیکربندی و نظارت بر مدیریت و تغییرات لازم در مدیریت.

۲. مدیریت پروژه و مدیریت بحران.

۳. بررسی و سنجیدن محیط. [۳]

## ۲-۴-۲-۲ آشنایی با نمودارهای UML

### ۲-۴-۲-۱-۲ دیاگرام مورد استفاده<sup>۱</sup>

این دیاگرام رفتار سیستم را توصیف می‌کند، که شامل تقابل بین سیستم و عاملان<sup>۲</sup> می‌باشد. الگوی رفتاری، توانایی‌های سیستم و یک دنباله تراکنش‌های وابسته به هم را توسط این دیاگرام نمایش می‌دهند. توصیفی از آنچه که سیستم انجام می‌دهد را از دید ناظر خارجی مدل‌سازی می‌کند و برای نمایش کارکرد کلی سیستم به کار می‌رود.

نکات مهم: ارتباط (عامل‌ها) با هم دیگر مجاز نیست. ضمناً ارتباط میان مورد کاربرها با خطوطی تحت عنوان include و یا extend مشخص می‌گردد. به هنگامی که یک مورد کاربر، دیگری را نیز شامل می‌شود از include و هنگامی که باعث گسترش مورد کاربر دیگری می‌شود از extend بهره می‌بریم.

در هنگام طراحی این نمودار توجه به این نکته مهم است که باید بیانگر فضای کلی پروژه باشد و نام‌گذاری هر مورد کاربر، به صورت واضح و مفهومی صورت بگیرد. در واقع موردهای کاربر همان عملیاتی هستند که در سیستم توسط کاربر(عامل) انجام می‌شود.

---

1-UseCase  
2- Actors

## ۲-۲-۴-۲ دیاگرام کلاس<sup>۱</sup>

این نمودار نوعی از ساختار ایستا می‌باشد که ساختار یک سیستم را با نمایش کلاس‌های سیستم، ویژگی‌ها و توانایی‌های آنها و همچنین ارتباطات<sup>۲</sup> بین آنها نشان می‌دهد.

در هر کلاس دو قسمت اصلی داریم: ویژگی‌ها که شامل خصوصیات آن عنصر برای شناسایی می‌باشد و توانایی‌ها که شامل قدرت و نحوه کاری است که عنصر در سیستم انجام می‌دهد. بنا به نیاز هر کدام از این قسمت‌ها در کلاس تعریف می‌شوند. ظاهر این نمودار به صورت جدولی شکل است و خط اول جدول نام کلاس و دو قسمت دیگر به ترتیب شامل ویژگی‌ها و توانایی‌ها می‌باشد.

## ۲-۲-۴-۳ دیاگرام توالی<sup>۳</sup>

در اصل این نمودار نوعی دیگر از دیاگرام تعاملی می‌باشد که نشانگر نحوه اجرای فرآیندها است، ضمناً ترتیب فعالیت‌ها را همان‌طور که از نامش پیداست متوالیاً نشان می‌دهد. مراحل انجام کار را مرحله به مرحله برای عملی شدن یک مورد کاربر نمایش می‌دهد. در رسم این نمودار، اصل بر درخواست اجرای کاری یا نمایش داده‌ای است. از این رو، شروع توالی فعالیت‌ها با درخواست از سمت کاربر همراه می‌باشد.

## ۲-۲-۴-۴ دیاگرام فعالیت<sup>۴</sup>

نمایش جریان کار و همچنین توالی فعالیت‌ها را ممکن می‌سازد. بسیار شبیه به فلوچارت می‌باشد. زیرا شما می‌توانید جریان کار را از یک فعالیت به فعالیت دیگر یا به حالتی دیگر، دنبال نمایید.

این دیاگرام شامل نقطه آغاز و پایان کار می‌باشد و همچنین در مواردی که بخواهیم نحوه انجام یک عملیات (فعالیت) را نشان بدهیم مورد استفاده قرار می‌گیرد. مثلاً وقتی یک فعالیت با خطا مواجه می‌شود و تراکنش قطع می‌شود، این نمودار توصیف می‌کند که چه پیش خواهد آمد.

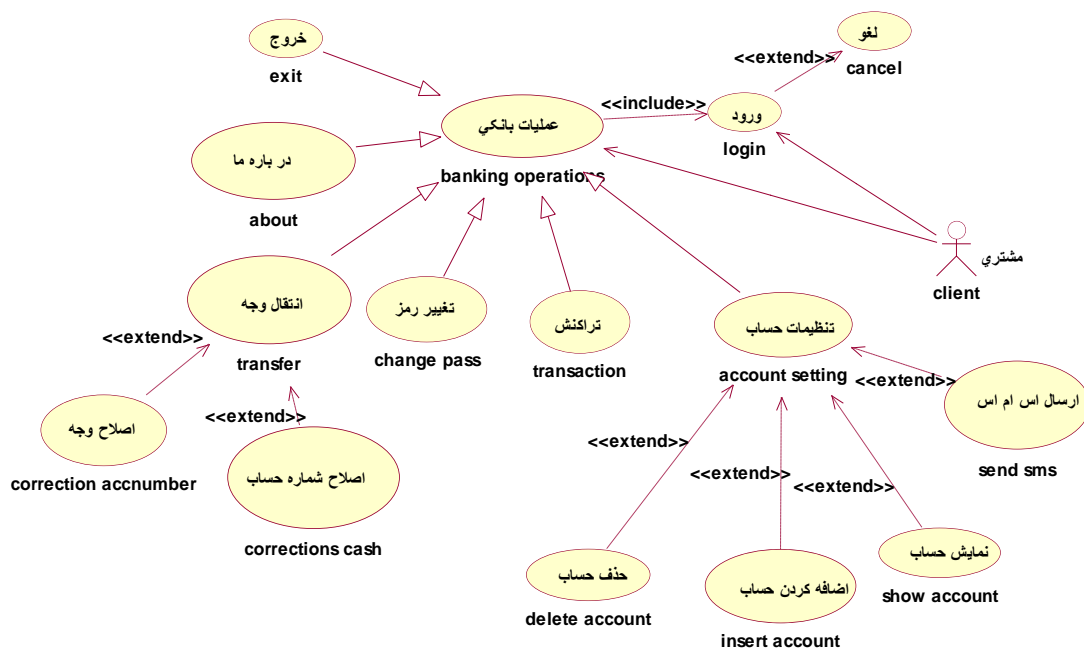
- 
- 1- Class Diagram
  - 2- Relationships
  - 3- Sequence Diagram
  - 4- Activity Diagram

عناصر مورد استفاده در ساخت این دیاگرام، عبارتند از:

- نقطه حالت آغازین<sup>۱</sup> و حالت نهایی<sup>۲</sup>
- خود فعالیت‌ها<sup>۳</sup> که نمایشگر مراحل اجرای کار هستند.
- گذار<sup>۴</sup> که مرحله تغییر و یا ترتیب کار را نمایش می‌دهد.
- شروط<sup>۵</sup> که شرایط تصمیم‌گیری را ممکن می‌سازند. [۴]

## ۲-۴-۳ نمودارهای مربوط به پروژه

### ۲-۴-۳-۱ نمودار مورد کاربر



شکل ۲-۲ نمایش نمودار مورد کاربر تولید شده در نرم‌افزار رشنال رز

- 
- 1- Start state
  - 2- End state
  - 3- activity
  - 4- transition
  - 5- decision

## ۲-۴-۳-۱-۱ جداول مربوط به سناریوهای نمودار کاربر

جدول ۲-۲ سناریو مورد کاربر ورود

۱		شماره:
	ورود	نام مورد کاربرد
	کاربر گوشی (مشتري)	عامل
	اعتبار سنجی شناسه و رمز کاربر برای ورود به سیستم	هدف
	کاربر برای ورود و انجام عملیات بانکی از شناسه و رمز استفاده می کند که از بانک دریافت کرده است. برای امنیت در انجام عملیات بانکی و دسترسی به پایگاه داده سرور، باید هویت کاربر شناسایی و صحت شناسه و رمز تأیید شود	نمای کلی
	اصلي	سطح
	کاربر باید از بانک شناسه و رمز عبور دریافت کند	پیش شرط
	کاربر وارد صفحه اصلی سیستم می شود و از سیستم اجازه ی انجام عملیات مورد نظر را دریافت می کند	شرط نهایی
	ندارد	مورد کاربرد مشمول (included)
	لغو عملیات	مورد کاربرد انتخابی (extended)
عملکرد سیستم	عملکرد عامل	سناریو موفق اصلی
	(۱) کاربر برنامه را اجرا می کند	
(۲) صفحه login نمایش داده می شود		
	(۳) کاربر شناسه و رمز را وارد می کند	
(۴) در صورت صحت داده های ورودی صفحه منو اصلی نمایش داده می شود		
عملکرد	شرط	سناریو نا موفق
سیستم پیام خطا در برقراری ارتباط را نمایش می دهد	ارتباط با سرور قطع باشد	
سیستم پیام غلط بودن شناسه یا رمز را می دهد	کاربر شناسه و رمز را اشتباه وارد کرده باشد	
	هر بار که برنامه اجرا می شود	تکرار

جدول ۲-۳ سناریو مربوط به مورد کاربر لغو عملیات

۲		شماره:
لغو عملیات		نام مورد کاربرد
کاربر گوشی (مشتری)		عامل
لغو عملیات ورود		هدف
کاربر از ورود به برنامه منصرف شده و از برنامه خارج میشود		نمای کلی
Extended		سطح
برنامه اجرا شده باشد		پیش شرط
کاربر از برنامه خارج میشود		شرط نهایی
ندارد		مورد کاربرد مشمول (included)
ندارد		مورد کاربرد انتخابی (extended)
عملکرد عامل	عمل کرد سیستم	سناریو موفق اصلی
(۱) کاربر برنامه را اجرا می کند		
(۲) صفحه login نمایش داده می شود		
(۳) کاربر منصرف شده و دکمه cancel را میزند		
(۴) سیستم صفحه خروج را نمایش می دهد و از کاربر تأیید کردن خروج را درخواست می کند		
(۵) کاربر خروج از برنامه را تأیید می کند		
هر بار که برنامه اجرا می شود		تکرار



جدول ۲-۴ سناریو مربوط به مورد کاربر عملیات بانکی

۳		شماره:
عملیات بانکی		نام مورد کاربرد
کاربر گوشی (مشتری)		عامل
انجام عملیات بانکی موجود در برنامه		هدف
کاربر میتواند عملیاتی چون انتقال وجه، نمایش حساب و موجودی... را از سیستم درخواست کند		نمای کلی
اصلی		سطح
کاربر باید وارد صفحه منو اصلی شده باشد		پیش شرط
وارد شدن به صفحات مربوطه برای انجام عملیات بانکی مد نظر		شرط نهایی
ورود		مورد کاربرد مشمول (included)
ندارد		مورد کاربرد انتخابی (extended)
عملکرد سیستم	عملکرد عامل	سناریو موفق اصلی
	(۱) کاربر بعد از ورود به برنامه و مشاهده منو یکی از عملیات را باید انتخاب کند	
(۲) سیستم با توجه به درخواست کاربر صفحه مدنظر را به کاربر نمایش میدهد		

جدول ۲-۵ سناریو مورد کاربر تنظیمات حساب

شماره:	۵
نام مورد کاربرد	تنظیمات حساب
عامل	کاربر گوشی (مشتری)
هدف	نمایش حساب‌ها و مشخصات آن‌ها، اضافه و حذف حساب
نمای کلی	کاربر می‌تواند شماره حساب جدید به سیستم اضافه و یا حذف کند، برای مشاهده موجودی و انجام تمامی عملیات‌های بانکی باید شماره حساب ثبت شده انتخاب شود
سطح	اصلی
پیش شرط	کاربر باید در صفحه منو قرار گرفته باشد
شرط نهایی	صفحه مد نظر را مشاهده می‌کند
مورد کاربرد مشمول (included)	ندارد
مورد کاربرد انتخابی (extended)	اضافه کردن حساب، حذف حساب، نمایش مشخصات حساب و اس ام اس
سناریو موفق اصلی	عملکرد عامل
	عمل کرد سیستم
	۱) کاربر از صفحه منو گزینه تنظیمات حساب را انتخاب می‌کند
	۲) سیستم صفحه مربوط به حساب‌ها را نمایش می‌دهد
۳) کاربر باید یکی از فعالیت‌ها را انتخاب کند (نمایش حساب، حذف حساب، نمایش مشخصات و یا اس ام اس)	
۴) سیستم به درخواست مشتری پاسخ می‌دهد و صفحه مورد نظر را باز می‌کند	

جدول ۲-۶ سناریو مربوط به مورد کاربر اضافه کردن حساب

۶		شماره:
اضافه کردن حساب		نام مورد کاربرد
کاربر گوشی (مشتری)		عامل
ثبت شماره حساب جدید در سیستم		هدف
در صورت نیاز کاربر شماره حساب جدید برای سیستم تعریف می کند		نمای کلی
Extended		سطح
کاربر در صفحه تنظیمات حساب باشد		پیش شرط
پیغام صحت در ثبت نمایش داده می شود		شرط نهایی
		مورد کاربرد مشمول (included)
		مورد کاربرد انتخابی (extended)
عملکرد سیستم	عملکرد عامل	سناریو موفق اصلی
	۱) کاربر اضافه شماره حساب را انتخاب می کند	
	۲) سیستم از کاربر نام و شماره حساب را درخواست می کند	
	۳) کاربر نام و شماره حساب را وارد کرده و دکمه ثبت را می زند	
	۴) سیستم در صورت دریافت اطلاعات و ثبت آنها پیغام تائید را نمایش می دهد	
عملکرد	شرط	سناریو نا موفق
سیستم پیغام می دهد که نام و شماره حساب را وارد نکرده اید	کاربر شماره حساب و نام حساب را وارد نکرده باشد و دکمه ثبت را بزند	

جدول ۲-۷ سناریو مربوط به مورد کاربر نمایش مشخصات حساب

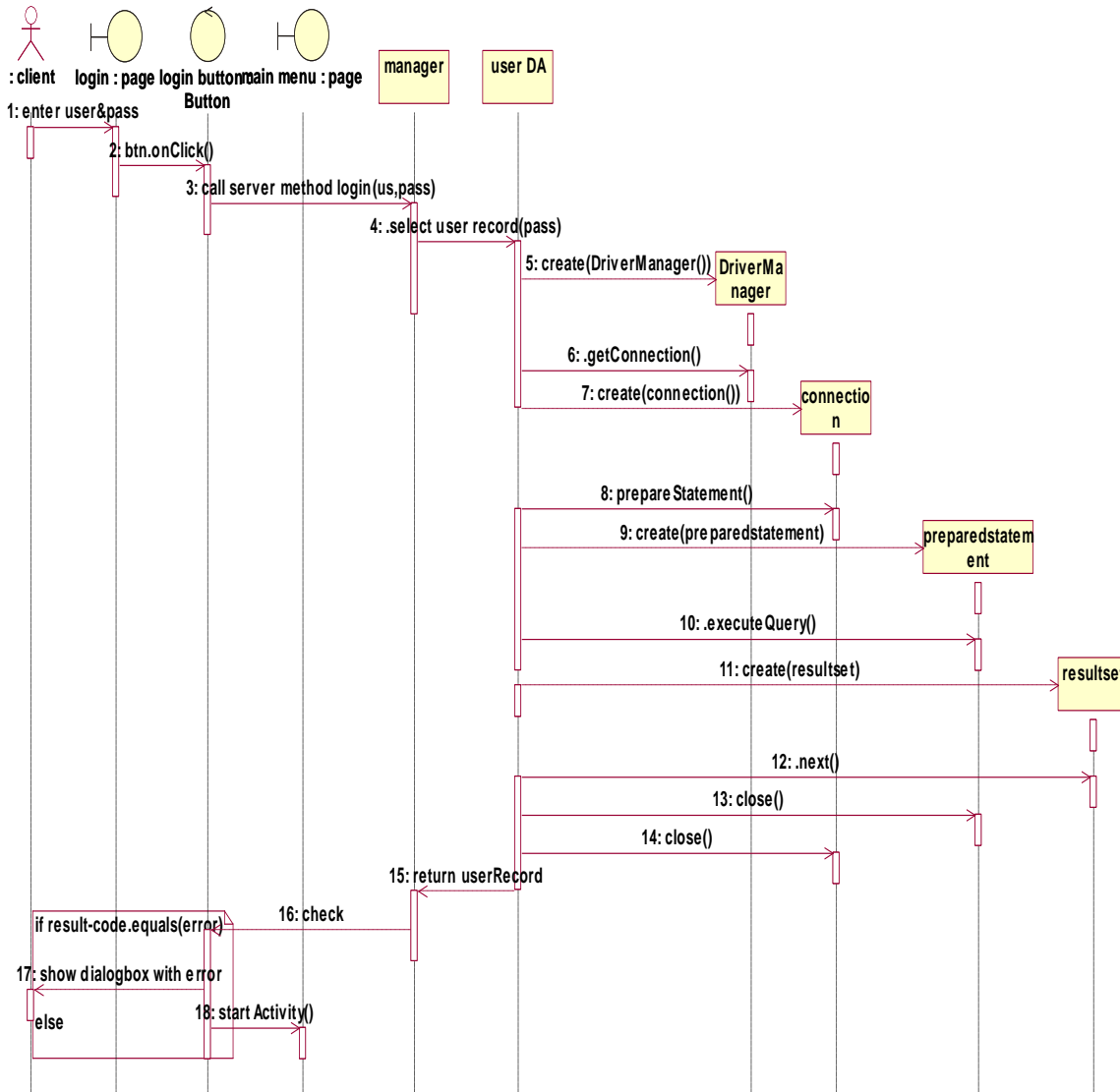
۷		شماره:
نمایش مشخصات حساب		نام مورد کاربرد
کاربر گوشی (مشتری)		عامل
نمایش موجودی حساب		هدف
کاربر با انتخاب کلید تنظیمات حساب وارد صفحه‌ای می‌شود که تمامی حساب-ها با مشخصات و موجودیشان قابل مشاهده می‌باشد		نمای کلی
Extended		سطح
کاربر باید در صفحه منو اصلی باشد		پیش شرط
		شرط نهایی
		مورد کاربرد مشمول (included)
		مورد کاربرد انتخابی (extended)
عملکرد عامل	عمل کرد سیستم	سناریو موفق اصلی
(۱) کاربر شماره حساب را انتخاب و دکمه نمایش را می‌زند		
	(۲) سیستم با توجه به شماره حساب وارد شده، به پایگاه داده سرور وصل، مشخصات و موجودی حساب را به کاربر نمایش می‌دهد	

جدول ۲-۸ سناریو مربوط به مورد کاربر انتقال وجه

شماره:	۸
نام مورد کاربرد	انتقال وجه
عامل	کاربر گوشی (مشتری)
هدف	انتقال وجه معینی از حساب فرد به حساب دیگران
نمای کلی	کاربر می‌تواند از هر کدام از حساب‌های خودش به فرد دیگری پول واریز کند
سطح	اصلی
پیش شرط	برای انجام این کار باید در سیستم شماره حسابی ثبت شده باشد
شرط نهایی	نمایش موفقیت در انتقال وجه
مورد کاربرد مشمول (included)	
مورد کاربرد انتخابی (extended)	اصلاح شماره مقصد، اصلاح وجه
سناریو موفق اصلی	عملکرد عامل
	عمل کرد سیستم
	(۱) کاربر از منو اصلی گزینه انتقال وجه را انتخاب می‌کند
	(۲) سیستم با نمایش دادن صفحه مربوطه از کاربر درخواست می‌کند که شماره حساب خود، شماره حساب مقصد و وجه را وارد کند
	(۳) کاربر بعد از وارد کردن مقادیر خواسته شده دکمه ادامه را می‌زند
	(۴) سیستم با توجه به مقادیر ورودی یک پیام نشان می‌دهد که شامل مشخصات گیرنده و مبلغ می‌باشد.
	(۵) کاربر بعد از مشاهده پیام اگر اطلاعات را صحیح وارد کرده باشد دکمه ارسال را می‌زند
	(۶) سیستم اطلاعات را به سرور می‌فرستد تا در صورت کافی بودن موجودی و صحیح بودن شماره

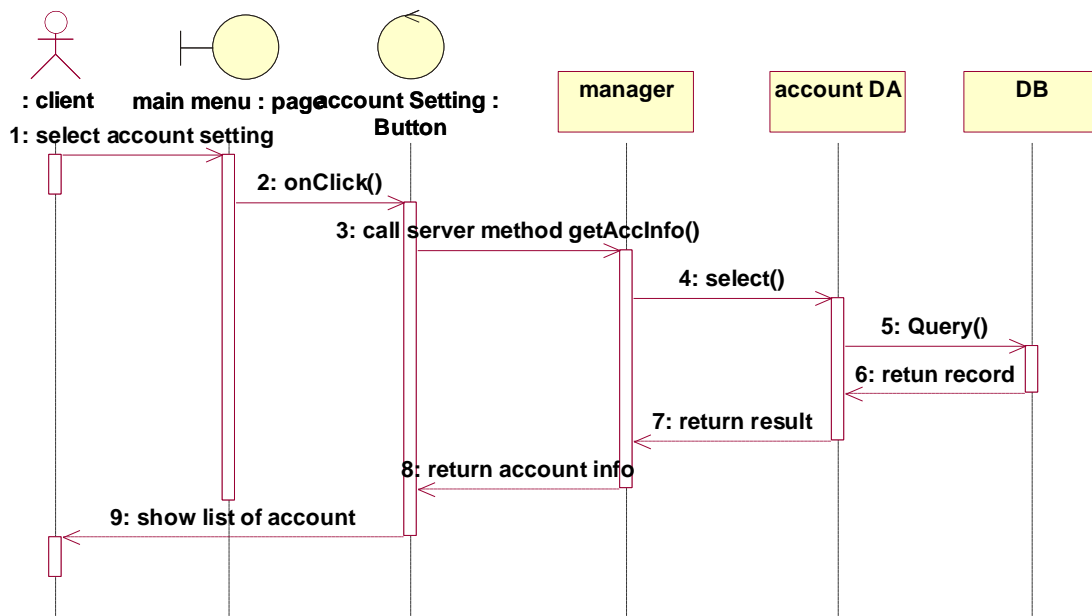
حساب مقصد عملیات انتقال انجام شود		
<b>عملکرد</b>	<b>شرط</b>	<b>سناریو ناموفق</b>
سیستم پیغام کافی نبودن موجودی را نمایش می دهد	موجودی کافی نباشد	
سیستم به کاربر اعلام می کند که شماره حساب مقصد نامعتبر است	شماره مقصد صحیح وارد نشده باشد	

۲-۳-۴-۲ نمودار توالی مربوط به ورود



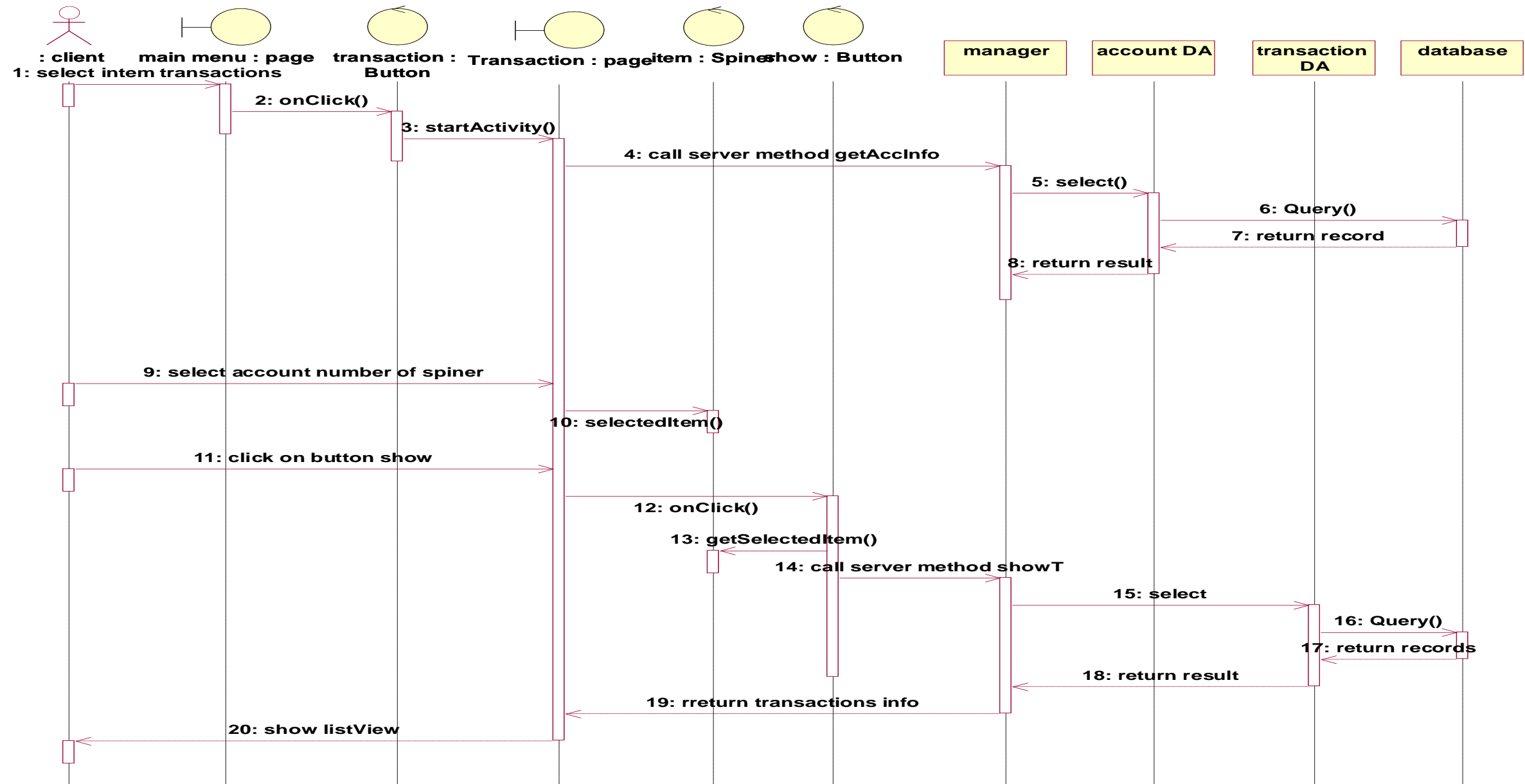
شکل ۲-۳ نمایش نمودار توالی ورود

۳-۳-۴-۲ نمودار توالی مربوط به مشاهده حساب



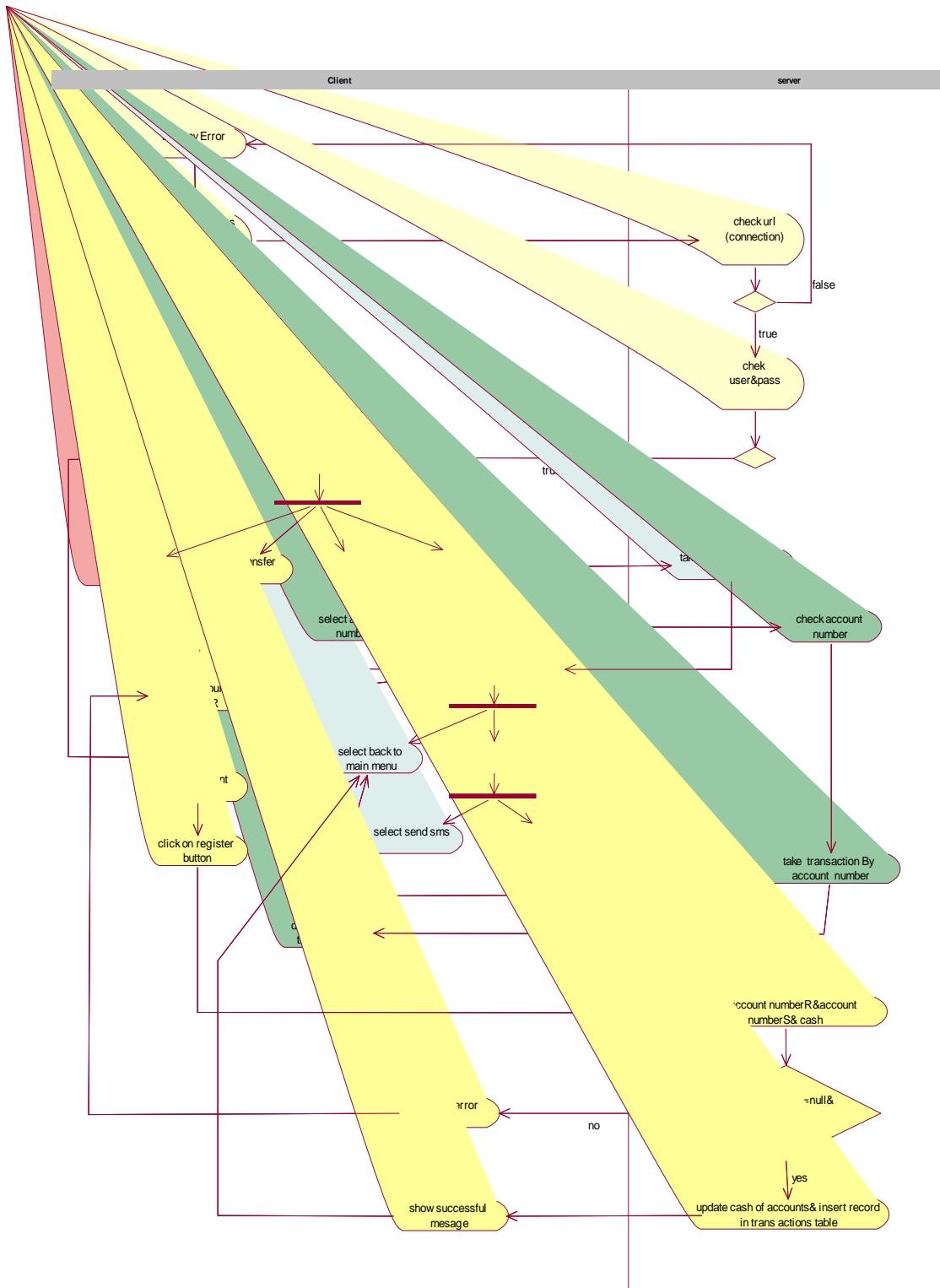
شکل ۴-۲ نمایش نمودار توالی مشاهده حساب





شکل ۲-۵ نمایش نمودار توالی مشاهده تراکنش‌ها

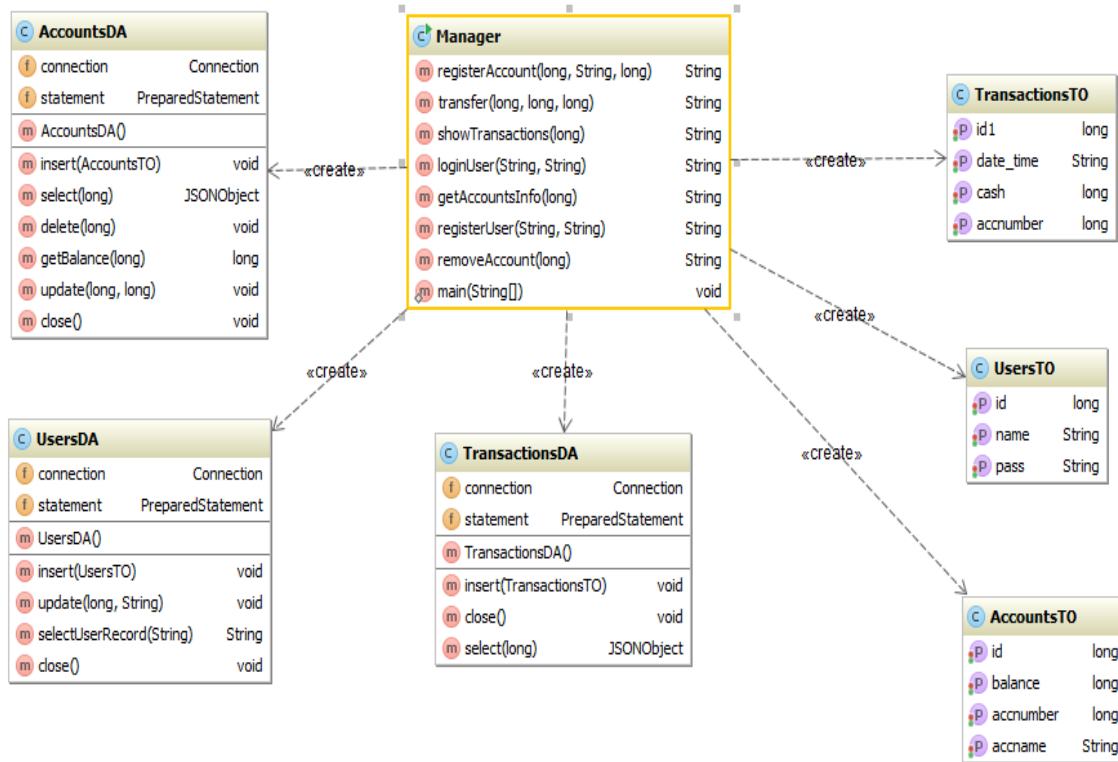
۵-۳-۴-۲ نمودار فعالیت



شکل ۶-۲ نمایش نمودار فعالیت

۶-۳-۴-۲ نمودار کلاس

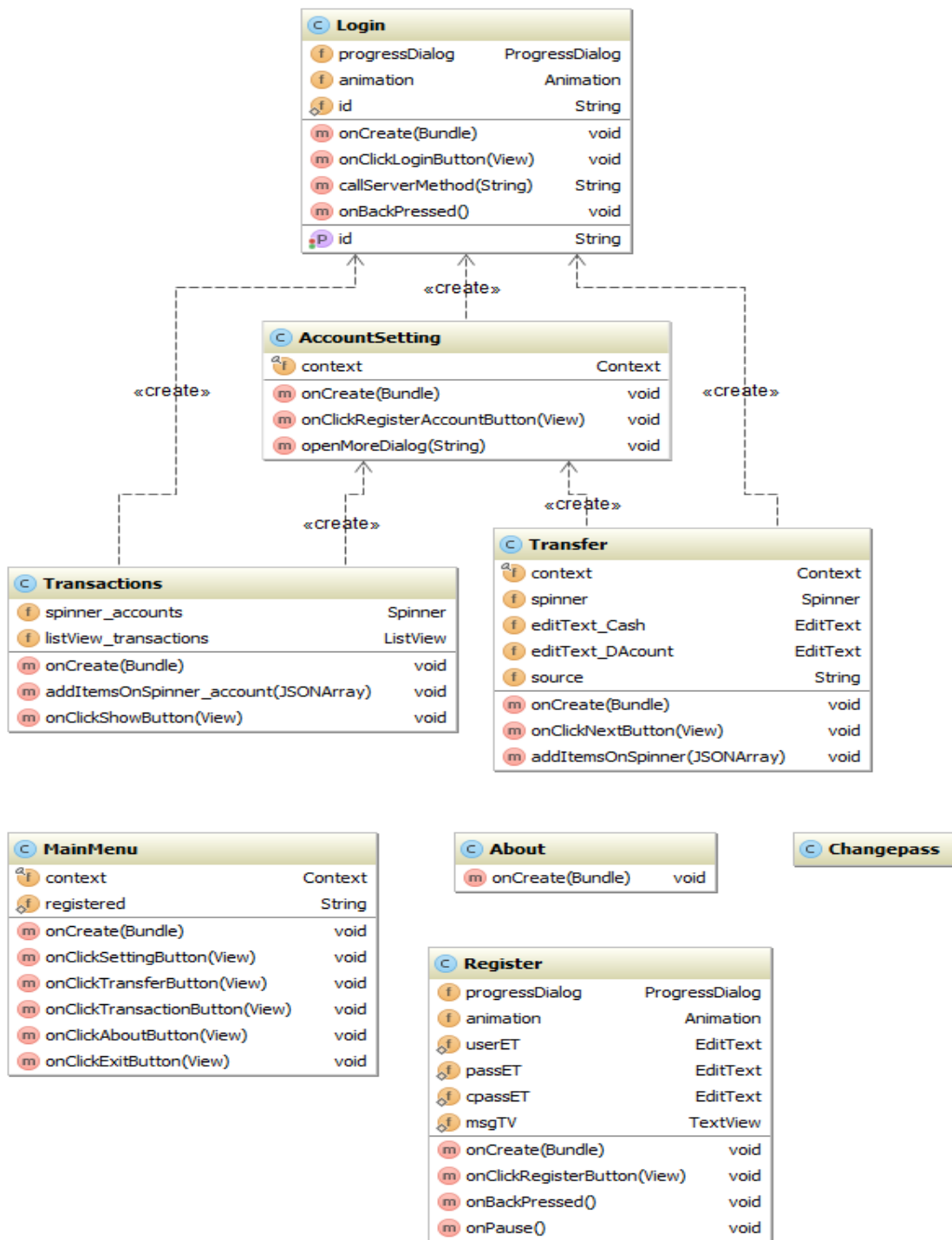
۱-۶-۳-۴-۲ نمودار کلاس سمت سرور



Powered by yFiles

شکل ۲-۷ نمودار کلاس سرور

۲-۴-۳-۶-۲ نمودار کلاس سمت کاربر



Powered by yFiles

شکل ۲-۸ نمودار کلاس کاربر

## فصل سوم

### سند چشم انداز

هدف از انجام این پروژه ایجاد برنامه کاربردی موبایل بانک تحت سیستم عامل اندروید است. علت انتخاب بستر اندروید برای پیاده سازی، گسترش روز افزون این سیستم عامل در انواع گوشی‌ها و تبلت‌ها میباشد. همچنین گسترش رقابت، همراه با تحولات وسیع در عرصه تجارت و بانکداری، بسیاری از روش‌های سنتی را متحول ساخته و فضای رقابت را در بکارگیری فن‌آوری‌های نو حاکم نموده است. نظام بانکی نیز از این قاعده مستثنی نبوده و با پیدایش مفاهیم نوین در بانکداری، شیوه ارائه خدمات به مشتریان در اقصی نقاط جهان دچار تحول شده است. فناوری بیسیم یا موبایل بانک شیوه نوینی از بانکداری است که توانسته نظر هر دو قشر خدمت دهنده و خدمت گیرنده را به خود جلب کند. این شیوه حتی در جهان هم نو محسوب میشود و هر روز نسخه‌های جدیدتر و کامل‌تری از آن وارد صنعت بانکداری دنیا میشود. در کشور ما هم این نوع از بانکداری نو ظهور است.

### ۲-۳ تاریخچه بازنگری (تجدید نظر)<sup>۱</sup>

#### ۱-۲-۳ جدول زمان بندی

جدول ۱-۳ زمان بندی تهیه سند

تاریخ	شرح	تهیه کننده
۹۲/۱۱/۲۰	تهیه سند اولیه	ساناز احمدزاده
۹۲/۱۲/۱۰	ویرایش سند	ساناز احمدزاده
۹۲/۱۲/۲۵	بازنگری و تأیید کلی سند	ساناز احمدزاده

#### ۲-۲-۳ معرفی پروژه

پروژه موبایل بانک اندروید از دو بخش تشکیل شده است. در بخش کاربر با زبان اندروید برنامه نوشته میشود و خروجی فایل قابل نصب<sup>۲</sup> بر روی گوشی است که شامل تمامی فعالیتها و صفحات گرافیکی برنامه می باشد. در قسمت سرور برنامه با زبان جاوا پیاده سازی می شود و هدف پیاده سازی سرور بانک به صورت مجازی است و به صورت فرضی پایگاه داده ای برای بانک با اوراکل طراحی شده است.

---

1- Revision History  
2 - APK

### ۳-۲-۳ محدوده

در این سند اطلاع‌رسانی دربارهٔ موبایل بانک و همچنین مراحل تولید برنامه و بخش‌های موجود در برنامه ذکر میشود. گستره این سند سیستم موبایل بانک تحت سیستم عامل اندروید و تمامی امور مربوط به آن را در بر می‌گیرد.

### ۳-۲-۴ تعاریف، اصطلاحات و اختصارات

جدول ۲-۳ تعاریف موجود در سند

واژه	شرح
موبایل بانک	شرکتی که فعالیت اصلی آن دریافت محصولات مختلف از تولیدکنندگان محصولات غذایی و سپس توزیع آن به مشتریان جزء می‌باشد.
کاربر	کلیه افرادی که در شرکت در جهت رسیدن به هدفی مشترک در کنار یکدیگر فعالیت می‌کنند.
اوراکل	نرم‌افزار طراحی و پیاده‌سازی پایگاه‌داده
GPRS	سرویس رادیویی بی سیم، جهت ارسال دیتا
سرور	منظور از سرور سیستم پاسخ دهنده به نیازهای کاربر می‌باشد

### ۳-۲-۵ منابع و مراجع

۱. مجلهٔ دانش و توسعه (علمی - پژوهشی) سال شانزدهم، شمارهٔ ۲۶، بهار ۱۳۸۸

تبیین موانع و راهکارهای توسعه بانکداری الکترونیکی " در چارچوب سند چشم انداز "ایران ۱۴۰۴



حمید آماده دانشیار دانشکده اقتصاد دانشگاه علامه طباطبایی، محمود جعفرپور دانشجوی کارشناسی

ارشد مدیریت بازرگانی دانشگاه علامه طباطبایی

۲. مجله شعبه مجازی شماره ۳۱ و ۳۲، تیر و مرداد ۸۷

### ۳-۳ موقعیت تجاری محصول

#### ۳-۳-۱ فرصت تجاری

با تولید برنامه موبایل بانک علاوه بر ایجاد سهولت در انجام کارهای بانکی برای مشتریان که با دلایلی همچون صرفه جویی در زمان، هزینه و دسترسی شبانه روزی موجب رضایتمندی آنان می‌شود، برای بانک‌ها هم سودآوری را به همراه دارد. ویژگی اصلی همه این فعالیت‌ها تسهیل فرآیندهای بانکی، کاهش هزینه‌های اداری به ویژه هزینه مکاتبات و بهبود دسترسی به مشتری است که بانک‌ها با افزایش تنوع خدمات ارائه شده به مشتریان سعی بر حفظ آنها دارند. براساس آمار بانک مرکزی بیش از ۷ میلیارد و ۱۱۰ میلیون قطعه اسکناس در ایران مورد استفاده قرار می‌گیرد که از این نظر ایران جایگاه اول را در دنیا در اختیار دارد از سویی عمر متوسط هر اسکناس در صورتی که به خوبی نگهداری شود ۳-۵ سال است اگر بخواهیم اسکناس‌های فرسوده را به موقع امحا کنیم باید هر ساله یک میلیارد و ۴۰۰ میلیون قطعه اسکناس از رده خارج شود که در حال حاضر تنها ظرفیت تولید ۷۰۰ میلیون قطعه وجود دارد. از سوی دیگر استقرار موبایل بانک موجب نزدیک شدن بانک‌ها به بانکداری الکترونیک میشود که علاوه بر ارتقاء سطح کیفی خدمات، ۶۰ تا ۷۰ درصد هزینه‌های بانک‌ها را کاهش می‌دهد مهمترین ویژگی‌ای که می‌توان برای بانکداری الکترونیک برشمرد، برقراری ارتباط سریع با تمام مخاطبان و ارائه خدمات مبتنی بر نیازهای فرد - فرد آنهاست.

### ۳-۳-۲ مشکلات

جدول ۳-۳ مسئله عدم اطلاع رسانی

مسئله	عدم اطلاع رسانی درباره وجود برنامه موبایل بانک به عموم
ذینفعان تأثیرپذیر	مدیران، مشتریان
تأثیرات منفی ناشی از مسئله	مراجعه حضوری به بانکها، زمان گیر و هزینه بر بودن
راه حل موفقیت آمیز	تبلیغات

جدول ۳-۴ مسئله شبکه و ارتباطات

مسئله	مشکلات شبکه و ارتباطات
ذینفعان تأثیرپذیر	مدیران، مشتریان برنامه نویس
تأثیرات منفی ناشی از مسئله	استفاده نکردن از برنامه و مراجعه حضوری به بانکها
راه حل موفقیت آمیز	رفع مشکلات پهنای باند و زیر ساخت‌های اینترنت و یا ایجاد قابلیت در برنامه که بتواند با پیام کوتاه اطلاعات را ارسال و دریافت کند برنامه نویس از شبیه ساز استفاده می کند برای تست و اجرا برنامه

جدول ۳-۵ مسئله امنیت

مسئله	امنیت
دینفعان تأثیرپذیر	مدیران و مشتریان
تأثیرات منفی ناشی از مسئله	عدم اعتماد به بانک واز دست دادن مشتریان. ضررهای مالی که به مشتریان وارد می شود
راه حل موفقیت آمیز	استفاده از شناسه و رمز عبور برای احراز هویت کد گذاری این اطلاعات و وارد کردن اطلاعات کد شده در پایگاه داده استفاده از راهای امنیتی در گوشی های همراه

جدول ۳-۶ مسئله تفاوت در نسخه های اندروید

مسئله	تفاوت بودن نسخه های اندروید و اندازه گوشی های موبایل
دینفعان تأثیرپذیر	مشتریان
تأثیرات منفی ناشی از مسئله	عدم استفاده از برنامه
راه حل موفقیت آمیز	مشخص کردن اینکه برنامه برای کدام نسخه طراحی شده

### ۳-۴ توصیف مشتری و ذینفعان

#### ۳-۴-۱ اطلاعات ذینفعان

جدول ۳-۷ اطلاعات ذینفعان

نام	توصیف	مسئولیت‌ها
تیم پروژه	مجری طرح	✓ تهیه برنامه زمان‌بندی انجام پروژه ✓ پیاده‌سازی پروژه ✓ پشتیبانی سیستم تهیه شده ✓ نظارت بر پیشرفت پروژه
سازمان یا موسسه‌های درخواست دهنده	کارفرما(بانک)	✓ ارزیابی راه کار انجام پروژه ✓ نظارت بر روند عملی پروژه ✓ مشخص کردن نیازمندی‌های سازمان ✓ تحویل‌گیری پروژه

#### ۳-۴-۲ اطلاعات مشتری

جدول ۳-۸ اطلاعات مشتری

نام	نقش	وظایف در پروژه
مشتریان بانک (کاربر)	استفاده کننده از برنامه جهت انجام امور بانکی	استفاده از برنامه

### ۳-۴-۳ محیط مشتری و ذینفع

جدول ۹-۳ جدول اطلاعات مربوط به محیط و مشتری

عنوان	شرح
سیستم عامل	اندروید
رابط کاربری	فرم‌های موجود در برنامه اندروید
ابزار ورودی	صفحه لمسی گوشی یا صفحه کلید آن
ابزار خروجی	مانیتور گوشی
محیط اجرا	گوشی و یا تبلت‌های اندروید
محدودیت استفاده از ابزار الکترونیکی در محیط	باید اینترنت و ایرلس موجود باشد

### ۳-۵ نیازمندیهای اصولی محصول

نصب نرم افزار IntelliJ IDEA12

نصب نرم افزار اوراکل ۱۱

### ۳-۶ نیازمندیهای محیطی

وجود بستری تحت سیستم عامل اندروید برای نصب فایل اجرایی برنامه

فصل چہارم

طراحی

## ۴-۱ مقدمه‌ای بر زبان برنامه نویسی اندروید

### ۴-۱-۱ تاریخچه اندروید<sup>۱</sup>

سیستم عامل<sup>۲</sup> اندروید برای نخستین بار توسط کنسرسیومی<sup>۳</sup> به رهبری گوگل معرفی شد، کمتر کسی پیش‌بینی می‌کرد که در این مدت کوتاه این سیستم عامل موفق به پیشی گرفتن از سیستم عامل‌های پرتعداد و جا افتاده تلفن همراه چون ویندوز موبایل، لینوکس شده و خود را به عنوان تهدیدی جدی برای رقابای چون سیمبین و آیفون نشان دهد. اندروید پا را از این هم فراتر گذاشته و علاوه بر حضور قدرتمند در بازار تلفن‌های همراه هوشمند، وارد عرصه‌های دیگری مانند تبلت‌ها و حتی تلویزیون نیز شده است.

در ماه ژوئیه سال ۲۰۰۵ گوگل شرکت اندروید در پالوآلتوی کالیفرنیا را خرید. شرکت کوچک اندروید که توسط اندی روبین، ریچ ماینرز، نیک سیرز و کریس وایت پایه‌گذاری شده بود، در زمینه تولید نرم‌افزار و برنامه‌های کاربردی برای تلفن‌های همراه فعالیت می‌کرد. اندی روبین مدیر ارشد اجرایی این شرکت پس از پیوستن اندروید به گوگل به سمت قائم‌مقام مدیریت مهندسی این شرکت و مسئول پروژه اندروید در گوگل منصوب شد. در واقع می‌توان روبین را پایه‌گذار اندروید دانست. چرا که وی علاوه بر اینکه ایده تولید اندروید را در شرکت کوچک خود پرورش داد، در سمت مدیر این پروژه در شرکت گوگل توانست ایده خود را پیاده‌سازی کند و سیستم عامل اندروید را با نام شرکت کوچک پیشین خود روانه بازار نماید. تیم اندروید به رهبری روبین فعالیت خود را برای تولید پلتفرم موبایل مبتنی بر کرنل لینوکس آغاز کردند. درز اخباری از فعالیت‌های این تیم به خارج از گوگل، سبب بروز شایعاتی مبنی بر تمایل گوگل به تولید تلفن همراه در اواخر سال ۲۰۰۶ گردید. این شایعات زمانی بیشتر قوت گرفت که در سپتامبر

---

1- Android  
2- Operating System  
3- Consortium

۲۰۰۷ نشریه اینفورمیشن ویک در گزارشی خبر از ثبت چندین حق امتیاز و اختراع در حوزه تلفن همراه توسط گوگل داد.

با اعلام زمان کنفرانس خبری شرکت گوگل در نوامبر سال ۲۰۰۷ دیگر تمامی رسانه‌ها و افکار عمومی جهان چشم انتظار مشاهده نخستین تلفن همراه ساخت گوگل بودند. ولی غافلگیری بزرگ رخ داد. هیچ خبری از «یک» گوشی تلفن همراه نبود بلکه خبر داغ آن روز در مورد ورود صدها تلفن همراه در سال‌های پیش رو بود که توسط شرکت‌های مختلف تولید می‌شد. در روز ۵ نوامبر ۲۰۰۷ اتحادیه‌ای اعلام موجودیت کرد، ۳۴ شرکت فعال در زمینه تولید نرم‌افزار، تولید گوشی‌های تلفن همراه، اپراتور تلفن همراه و تولید کننده نیمه رساناها و پردازنده‌های تلفن همراه اعضای مؤسس این اتحادیه بودند. در میان نام‌های مشهور در بین اعضای مؤسس، شرکت‌هایی چون سامسونگ، موتورولا، اینتل، تگزاس اینسترومنتس، کوآلکام، برادکام، تلفونیکا، اسپرینت و البته گوگل به چشم می‌خوردند. اریک اشمیت مدیر ارشد اجرایی گوگل در این مراسم گفت: «اعلام امروز بسیار جاه‌طلبانه‌تر از معرفی تنها «یک» تلفن گوگلی است که در چند هفته اخیر توسط رسانه‌ها پیش‌بینی شده بود. از دیدگاه ما پلتفرمی که ما ارائه کرده‌ایم، هزاران تلفن گوناگون را به بازار روانه خواهد کرد.» نخستین گوشی مبتنی بر آندروید توسط شرکت HTC تولید شد.

#### ۴-۱-۲ کپی‌رایت و حق امتیاز

حق امتیاز<sup>۱</sup> آندروید به صورت منبع باز<sup>۲</sup> بر اساس حق امتیاز آپاچی<sup>۳</sup> ارائه می‌گردد. بر این اساس شرکت‌های عضو اتحادیه می‌توانند با دسترسی به کدهای اصلی آندروید آن را مطابق دلخواه خود تغییر دهند و کد تغییر یافته را بدون عودت دادن برای خود حفظ کنند.

---

1- License  
2- Open Source  
3- Apache License



#### ۴-۱-۳ سهم بازار اندروید

سهم بازار اندروید در مقایسه با سایر سیستم‌های عامل تلفن‌های هوشمند، رشد اعجاب‌آور اندروید را نشان می‌دهد. اندروید برای نخستین بار در سه ماهه اول سال ۲۰۱۰ توانست گوشی‌های بیشتری از مهم‌ترین رقیب خود یعنی اپل به فروش برساند. برخی کارشناسان بر این باور هستند که اگر گوگل موفق شود اپل را از پیش روی خود بردارد سیمبین را نیز پشت سر خواهد گذاشت. گروهی از کارشناسان، استراتژی اندروید در مقابله با اپل را با استراتژی مایکروسافت در اوایل دهه ۱۹۷۰ مقایسه می‌کنند. جایی که مایکروسافت توانست با فروش حق امتیاز استفاده از سیستم عامل خود به سایر شرکت‌ها به سلطه مکتینتاش خاتمه دهد و حالا گوگل به همین استراتژی و به کمک بزرگ‌ترین تولید کنندگان تلفن همراه مانند سامسونگ، ال‌جی، سونی اریکسون، موتورولا و اچ تی سی، قصد دارد روند رشد آیفون اپل را متوقف سازد و به نظر می‌رسد تا حد زیادی هم موفق بوده است. بر اساس آمار ارائه شده توسط شرکت Admob در آوریل ۲۰۱۰ تعداد کل گوشی‌های آیفون موجود در بازار ایالات متحده آمریکا ۱۰/۷ میلیون دستگاه بوده است. این در حالی است که تعداد گوشی‌های مبتنی بر اندروید ۸/۷ میلیون دستگاه بوده است. ذکر این نکته ضروری است که نخستین گوشی آیفون در ۲۹ ژوئن ۲۰۰۷ به بازار عرضه شد در حالی که نخستین گوشی مبتنی بر اندروید بیش از یک سال بعد و در اکتبر ۲۰۰۸ روانه بازار شد. اما به غیر از اپل بقیه رقبا نیز از دست اندروید جان به در نبرده‌اند. اندروید در سه ماهه نخست سال ۲۰۱۰ توانست سهم بازار خود را از ۱/۶ درصد در مدت زمان مشابه در سال گذشته به ۹/۶ درصد برساند و با پشت سر گذاشتن ویندوز موبایل و لینوکس در رده چهارم پرطرفدارترین سیستم عامل تلفن‌های همراه هوشمند قرار گیرد.

با اقبال بیشتر سایر تولید کنندگان به گوشی‌های اندروید به نظر می‌رسد روند رشد این سیستم عامل نه تنها کند نگردد بلکه شتاب بیشتری نیز پیدا کند. تاکنون بالغ بر ۶۱ مدل دستگاه مبتنی بر اندروید با ۲۱ برند مختلف تولید شده است. بنابر آخرین گزارش‌ها در حال حاضر هر روز یکصد هزار گوشی مبتنی بر

آندروید به فروش می‌رسد. با نرخ کنونی گوگل ۳۶ میلیون گوشی در سال به فروش خواهد رساند. این رقم زمانی معنا پیدا می‌کند که بدانیم شرکت اچ تی سی، چهارمین تولید کننده تلفن‌های همراه هوشمند در جهان سالانه ۱۷ میلیون گوشی تلفن همراه به فروش می‌رساند. با این اوصاف انتظار می‌رود آندروید بتواند جهشی شگرف در سهم بازار را رقم زند.

## ۴-۲ اندروید مارکت چیست؟

آندروید مارکت سرویس فروش نرم‌افزارهای کاربردی برای گوشی‌های آندروید است. یک برنامه کاربردی ویژه آندروید مارکت به صورت از پیش بارگذاری شده بر روی گوشی‌های آندروید نصب گردیده و به کاربران امکان می‌دهد نرم‌افزارهای مورد نیاز خود را خریداری و دانلود کنند. البته تمامی نرم‌افزارهای موجود در آندروید مارکت فروشی نیستند بلکه بیش از نیمی از نرم‌افزارهای موجود در آندروید مارکت به صورت رایگان عرضه می‌شوند و از این نظر آندروید بیشترین درصد نرم‌افزارهای رایگان را در بین تمامی سیستم‌های عامل تلفن‌های همراه هوشمند در اختیار کاربران قرار می‌دهد. هر برنامه‌نویس با ثبت نام، امکان فروش برنامه‌های خود در آندروید مارکت را دارد. ۷۰ درصد از مبلغ فروش برنامه‌های کاربردی به برنامه‌نویس تعلق می‌گیرد و ۳۰ درصد مابقی بین اپراتورها توزیع می‌شود. بر اساس سیاست‌های گوگل در حال حاضر تمامی برنامه نویسان عضو پروژه آندروید از سراسر جهان می‌توانند برنامه‌های کاربردی رایگان خود را از طریق آندروید مارکت در ۴۶ کشور عرضه کنند. برای اینکار کافی است برنامه نویسان فرمی مختصر را تکمیل کرده و البته ۲۵ دلار حق عضویت هم به گوگل بپردازند. ولی تنها برنامه‌نویسان ساکن در نه کشور اتریش، فرانسه، آلمان، ایتالیا، ژاپن، هلند، اسپانیا، انگلستان و ایالات متحده آمریکا می‌توانند برنامه‌های خود را برای فروش در ۱۳ کشور استرالیا، اتریش، کانادا، فرانسه، آلمان، ایتالیا، ژاپن، هلند، نیوزلند، اسپانیا، سوئیس، انگلستان و ایالات متحده آمریکا در معرض بازدید خریداران قرار دهند و سایر کشورها امکان مشاهده و خرید برنامه‌های غیر رایگان را ندارند. [۵]

## ۳-۴ زبان برنامه‌نویسی اندروید

زبان برنامه‌نویسی اصلی این سیستم عامل زبان جاوا می‌باشد که به عنوان پرکاربردترین زبان برنامه‌نویسی در دنیا است و به طور حتم علت انتخاب این زبان برنامه‌نویسی برای سیستم عامل اندروید این می‌باشد که جاوا بسیار قدرتمند بوده و در عین حال Open Source نیز می‌باشد.

## ۴-۴ اجزای تشکیل دهنده سیستم‌عامل اندروید

سیستم عامل اندروید بر پایه سیستم عامل لینوکس طرح ریزی شده است و ماشینی مجازی<sup>۱</sup> در این سیستم عامل تحت عنوان Dalvik تعبیه شده است که برای دستگاه‌های موبایل بهینه‌سازی شده است. به طور خلاصه می‌توان گفت که این ماشین مجازی دارای وظایف متعددی از جمله مدیریت حافظه، فشرده‌سازی بیشتر اپلیکیشن و سازگاری با واحدهای پردازش مرکزی مختلف در انواع مختلف دستگاه‌ها بدون نیاز به بازنویسی اپلیکیشن برای هر واحد پردازش مرکزی خاص، می‌باشد.

برای جستجو در اینترنت، سیستم عامل اندروید از موتور جستجوی<sup>۲</sup> WebKit استفاده می‌کند. گرافیک آن با OpenGL ES بهینه‌سازی شده و برای ذخیره‌سازی اطلاعات از پایگاه داده SQLite استفاده می‌کند. همانطور که درمباحث قبلی اشاره شد، زبان برنامه‌نویسی اصلی سیستم عامل اندروید جاوا است. در واقع پس از تکمیل یک اپلیکیشن اندروید، این اپلیکیشن در قالب یک فایل با پسوند<sup>۳</sup> apk کامپایل می‌شود که این فایل را می‌توان در بازارهای مختلفی همچون Google Play و غیره در اختیار علاقمندان قرار داد تا بتوانند با نصب آن از اپلیکیشن‌ها استفاده نمایند.

---

1- Virtual machine  
2- Search Engine  
3- Application Package File

## ۴-۵ محیط‌های برنامه نویسی اندروید

### ۴-۵-۱ محیط برنامه نویسی اکلیپس

معروف‌ترین محیط برنامه نویسی برای توسعه اندروید محیط برنامه نویسی اکلیپس<sup>۱</sup> به همراه ابزارهای SDK<sup>۲</sup> و ADT<sup>۳</sup> است. در واقع از آنجا که محیط برنامه نویسی اکلیپس متن باز است شرکت گوگل تمایل بیشتری برای فراهم آوردن ابزارهای توسعه اندروید برای این محیط برنامه نویسی قدرتمند از خود نشان داده‌است.

### ۴-۵-۱-۱ نسخه ADT Bundle محیط برنامه نویسی اکلیپس

این محیط برنامه نویسی دقیقاً همان محیط برنامه‌نویسی اکلیپس است با این تفاوت که خود شرکت گوگل تغییراتی در آن صورت داده و آن را منتشر کرده است. این محیط برنامه نویسی بر خلاف اکلیپس که برای توسعه اندروید می‌بایست تنظیمات خاصی در آن انجام داد، در برگیرنده کلیه ابزارهای لازم برای توسعه اندروید از جمله SDK، ADT و شبیه ساز<sup>۴</sup> بوده و کاربران دیگر نیازی به دانلود کردن این ابزارها به صورت مجزا نخواهند داشت. لازم به ذکر است توسعه دهندگانی که با نحوه کار با محیط برنامه نویسی اکلیپس آشنایی داشته باشند به راحتی خواهند توانست از این محیط هم در تولید اپلیکیشن اندروید استفاده کنند.

### ۴-۵-۲ محیط برنامه نویسی اندروید استودیو

اندروید استودیو<sup>۵</sup> محیطی برای توسعه برنامه‌های اندرویدی است که برای اولین بار خود شرکت گوگل بر پایه محیط برنامه نویسی قدرتمند IntelliJ IDEA طراحی کرده است که همانند اکلیپس ابزارهای لازم برای طراحی، توسعه و مشکل‌یابی برنامه‌ها را داراست. اولین نسخه از اندروید استودیو در کنفرانس سالیانه گوگل که در اردیبهشت ماه سال ۱۳۹۱ اتفاق افتاد تحت عنوان نسخه ۱ معرفی شد.

---

1-Eclipse

2-Software Development Kit

3-Android Development Tools

4- Emulator

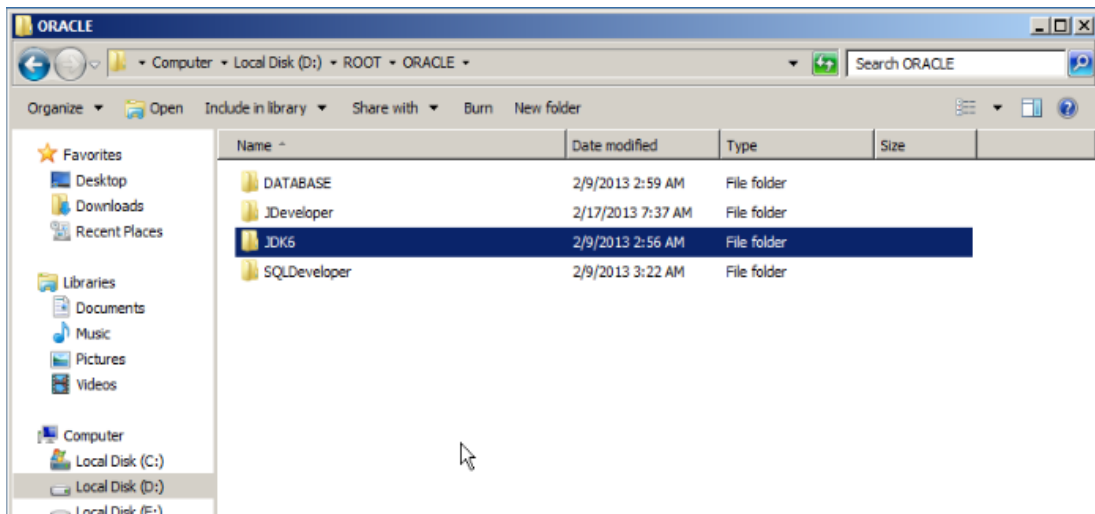
5- Android Studio

## ۳-۵-۴ محیط برنامه‌نویسی IntelliJ IDEA

### ۱-۳-۵-۴ مراحل نصب محیط IntelliJ

(۱) نصب کیت توسعه جاوا<sup>۱</sup>

ابتدا فایل JDK را در درایو غیرسیستمی کپی می‌کنیم.



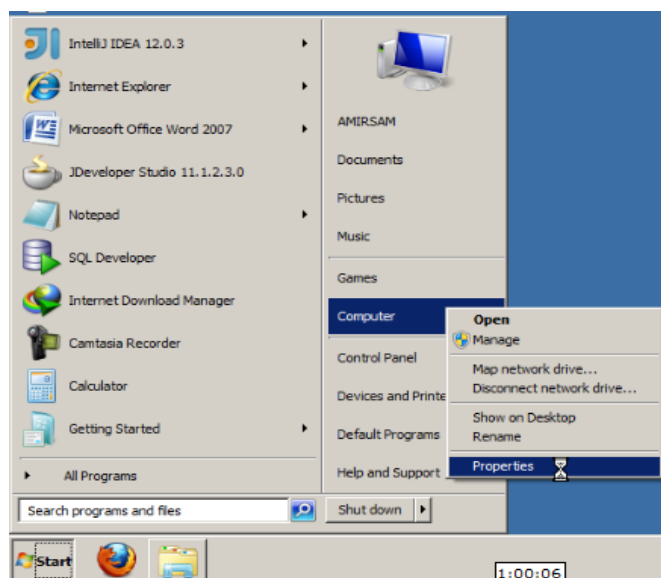
شکل ۱-۴ نمایش محل کپی کردن فایل JDK

(۲) انجام دادن تنظیمات مربوط به متغیرهای سیستم عامل:

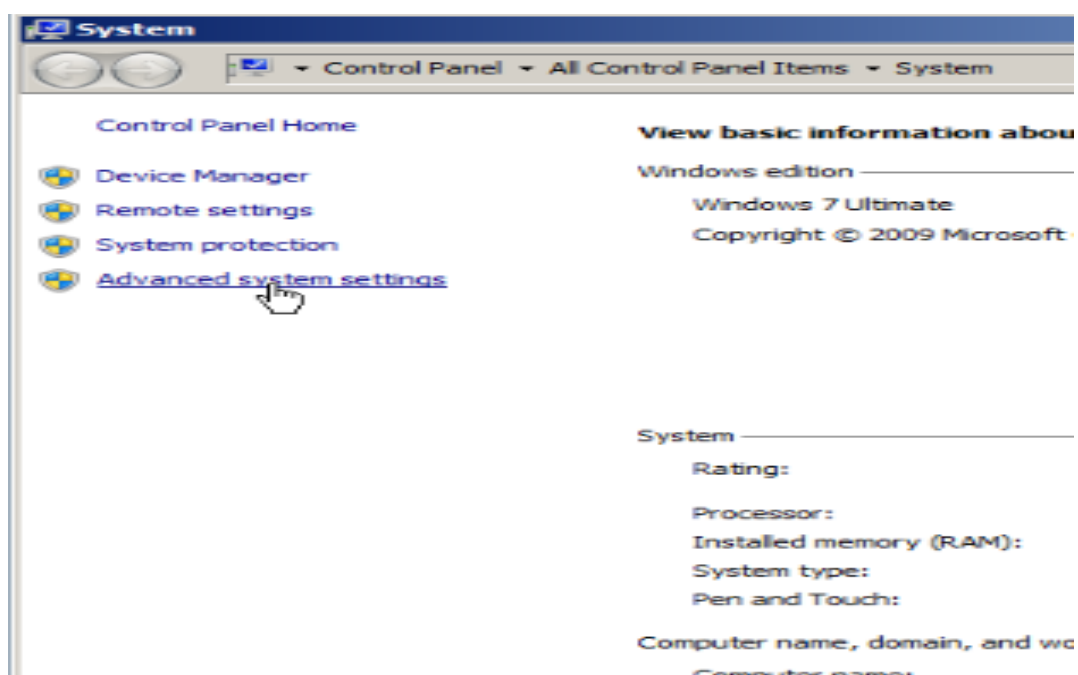
برای انجام دادن این کار از منو شروع<sup>۲</sup> بر روی آیکون کامپیوتر راست کلیک کرده و Properties را انتخاب می‌کنیم در ادامه از صفحه تنظیمات پیشرفته<sup>۳</sup> بر روی متغیرهای محیطی<sup>۴</sup> کلیک کرده و از قسمت متغیرهای سیستمی<sup>۵</sup> با انتخاب path در پنجره ایجاد شده عبارت زیر را تایپ می‌کنیم.

%JAVA\_HOME%\bin

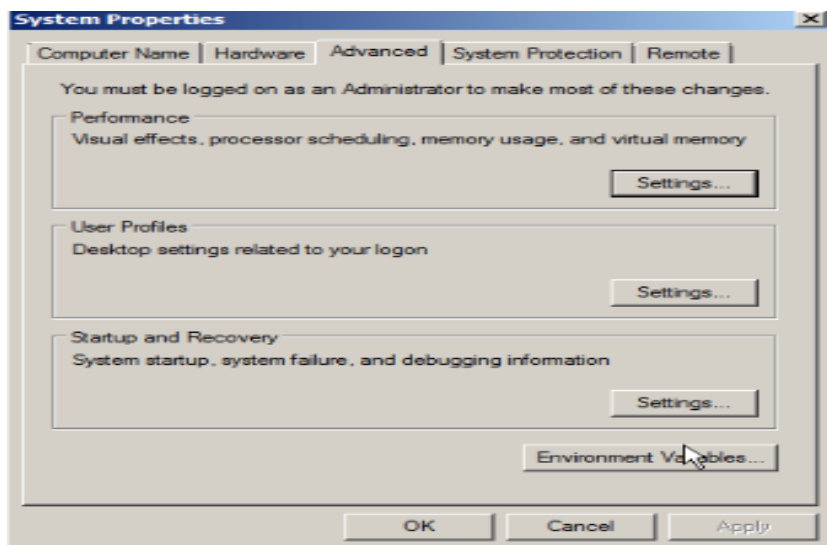
- 
- 1-JDK6
  - 2- Start
  - 3- Advanced Setting
  - 4- Environment Variables
  - 5- System Variables



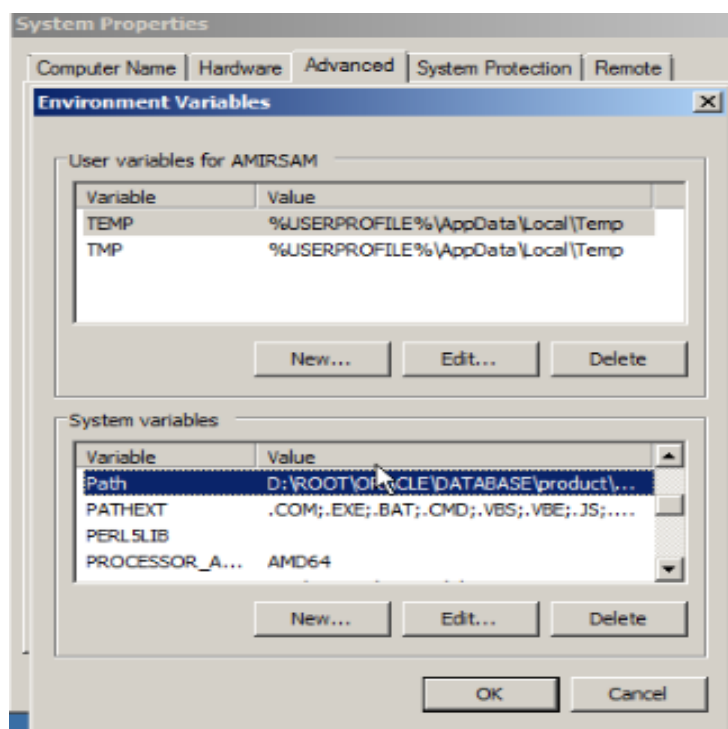
شکل ۴-۲ نمایش مرحله اول تنظیمات متغیرهای سیستمی



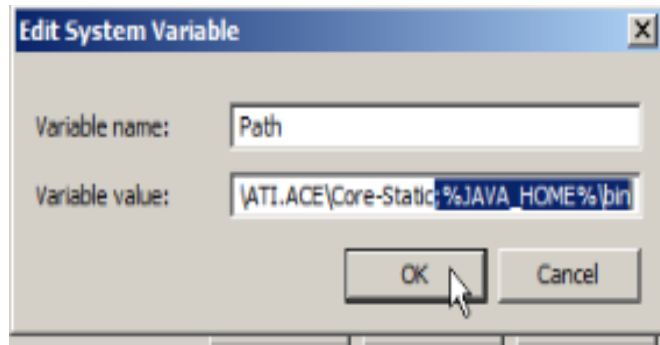
شکل ۴-۳ نمایش مرحله دوم تنظیمات متغیرهای سیستمی



شکل ۴-۴ نمایش مرحله سوم تنظیمات متغیرهای سیستمی

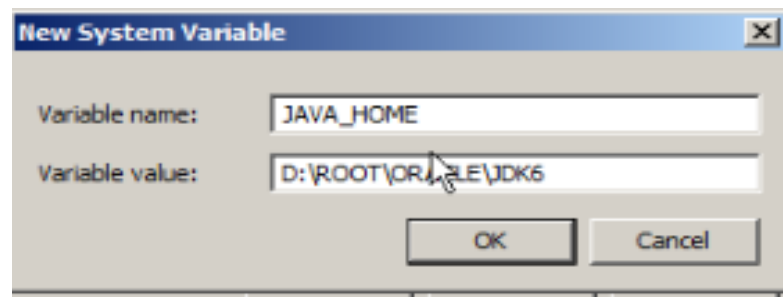


شکل ۴-۵ نمایش مرحله چهارم تنظیمات متغیرهای سیستمی



شکل ۴-۶ نمایش مرحله پنجم تنظیمات متغیرهای سیستمی

بعد از ok کردن در قسمت system variables ، new را زده و در پنجره ظاهر شده آدرس جایی که JDK را ذخیره کرده بودیم می نویسیم .



شکل ۴-۷ نمایش مرحله ششم تنظیمات متغیرهای سیستمی

برای اطمینان از نصب درست JDK در cmd دستور `JAVA -version` را وارد می کنیم که نسخه jdk را نمایش می دهد.



```
C:\Windows\system32\CMD.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\AMIRSAM>JAVA -version
java version "1.6.0_12"
Java(TM) SE Runtime Environment (build 1.6.0_12-b04)
Java HotSpot(TM) Client VM (build 11.2-b01, mixed mode)

C:\Users\AMIRSAM>
```

شکل ۴-۸ مربوط به نمایش اطلاعات مربوط به کیت توسعه جاوا

۳) نصب IntelliJ IDEA .

## ۴-۶ معرفی فایل‌های موجود در برنامه‌های اندروید

به طور کلی هر برنامه شامل فایل‌های جزئی<sup>۱</sup>، فایل‌های ظاهری<sup>۲</sup> و منابع<sup>۳</sup> می‌باشد.

### ۴-۶-۱ معرفی فایل‌های جزئی

فایل‌های جزئی شامل موارد زیر هستند:

- فعالیت‌ها<sup>۴</sup>
- خدمات<sup>۵</sup>

- 
- 1- Component
  - 2- Manifest
  - 3- Resources
  - 4- Activities
  - 5- Services

- ارائه دهنده‌ی محتوا<sup>۱</sup>

- گیرنده‌ی پخش<sup>۲</sup>

#### ۴-۶-۱-۱ معرفی فعالیت‌ها

اگر بخواهیم معنای یک فعالیت را در سیستم عامل اندروید به خوبی متوجه شویم، می‌توان آن را به منزله یک صفحه از اپلیکیشن<sup>۳</sup> تصور کرد. برای روشن شدن این مسئله مثالی ذکر می‌کنیم. فرض کنیم که یک اپلیکیشن را اجرا می‌کنیم، این اپلیکیشن پس از اجرا وارد صفحه اصلی برنامه می‌شود. این صفحه اصلی یک فعالیت است. حال می‌بینیم که چندین دکمه در این صفحه اصلی برای رفتن به بخش‌های مختلف برنامه تعبیه شده به طور مثال یک دکمه جهت صفحه راهنما، یک دکمه جهت ارتباط با طراح اپلیکیشن، یک دکمه جهت وارد شدن به موتور جستجو در اینترنت و غیره، زمانیکه ما پس از زدن دکمه راهنما وارد صفحه راهنما می‌شویم در واقع وارد یک فعالیت دیگر شده ایم. اکنون این فعالیت جدید فعالیت قبلی که مربوط به صفحه اصلی برنامه بود را پس زده و جای آن را می‌گیرد. در این حین اگر دکمه بازگشت تلفن همراه خود را فشار دهیم، فعالیت قبلی مجدد بالا آمده و فعالیت مربوط به صفحه راهنما را پس می‌زند.

اکنون ببینیم که یک فعالیت را به چه نحوی می‌توان ساخت. در حقیقت دو راه برای ساخت فعالیت‌ها در سیستم عامل اندروید وجود دارد: راه اول که به صورت دینامیک و پویا است با استفاده از برنامه نویسی جاوا می‌باشد و راه دوم با استفاده از XML است. می‌توانید طول عمر یک فعالیت را با توابع مختلفی زیر نظر داشته باشید، توابعی مثل `onStart()` یا `onPause()`، تا روند اجرای هر فعالیتی را کنترل کنید.

حال به بررسی کلاس‌های فعالیت می‌پردازیم برای این کار در پوشه‌ی `src` روی کلاس `MyActivity`

کلیک می‌کنیم تصویر زیر کدهای موجود در این کلاس را نشان می‌دهد :

---

1- Content Provider  
2- Broadcast Receiver  
3- Application

```

package com.example.untitled9;

import android.app.Activity;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;

public class MyActivity extends Activity {
    /**
     * Called when the activity is first created.
     */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);
    }
}

```

شکل ۴-۹ کدهای مربوط به کلاس فعالیت

اولین خط از کد فوق مربوط به نام پکیجی است که در حین ساخت پروژه تعریف می‌کنیم. در واقع وظیفه این پکیج جای دادن کلیه فایل‌های جاوا است که در حین ساخت یک اپلیکیشن ایجاد خواهیم کرد، پس از این خط دو خط کد دیگر وارد برنامه شده است که با کلید واژه import آغاز شده‌اند. به طور کلی این دو خط کد وظیفه دارند تا پکیج‌های مورد نیاز برای توسعه یک اپلیکیشن را به پروژه ما import یا "وارد" کنند.

در حقیقت کلاس فعالیت به عنوان مهم‌ترین کلاس در ساخت یک اپلیکیشن اندرویدی است زیرا همانطور که قبلاً توضیح داده شد در یک اپلیکیشن اندرویدی هر چیزی با یک فعالیت آغاز می‌شود و از همین رو است که کلاس اصلی که برای یک اپلیکیشن در حین ساخت یک پروژه تعریف می‌کنیم خصوصاً را از کلاس فعالیت به ارث می‌برد. به عبارت دیگر کد `MainActivity extends Activity` حاکی از آن است که کلاس `MainActivity` که در پروسه ساخت پروژه خود ایجاد کردیم برگرفته از کلاس فعالیت است، به عبارت دیگر از کلاس فعالیت که توسط برنامه نویسان سیستم عامل اندروید نوشته شده است ارث بری می‌کند.

از آنجا که ما نیاز داریم تا بارها و بارها از کلاس فعالیت در اپلیکیشن خود استفاده کنیم، نیاز است تا این کلاس را از طریق کد `import android.app.Activity;` به پروژه خود وارد کنیم (لازم به ذکر است که می توانیم این کلاس را وارد برنامه نکرده و هر جا که خواستیم از این کلاس استفاده کنیم آدرس کامل آن را بنویسیم. به طور مثال در این پروژه می توانیم کد `MainActivity extends android.app.Activity;` را به اپلیکیشن خود اضافه کنیم اما این در حالی است که با اتخاذ این رویکرد می بایست کد بیشتری بنویسم که به نظر می رسد این کار منطقی نباشد.

@Override

```
public void onCreate(Bundle savedInstanceState) {  
  
    super.onCreate(savedInstanceState);  
  
    setContentView(R.layout.main);  
  
}
```

کلاس فعالیت در برگیرنده متدهای فراوانی است که یکی از آنها همین متد `onCreate` است که در کد فوق مشاهده می شود. به طور کلی می توان گفت زمانیکه یک فعالیت ساخته می شود، چندین متد فراخوانده می شوند که اولین آنها متد `onCreate` است که با استفاده از آن خواهیم توانست هرکاری که تمایل داشته باشیم را اجرا کنیم. به طور مثال در این فعالیت پس از فرا خواندن این متد قادر خواهیم شد تا رابط گرافیکی کاربر<sup>1</sup> برنامه را که در فولدر `layout` زیرمجموعه فولدر `res` می باشد و نام آن `main.xml` است را اجرا نماییم. در زبان برنامه نویسی جاوا می توانیم متدهای مرتبط با کلاسی مثل کلاس فعالیت که کلاس دیگری مثل

---

1- GUI

کلاس `MainActivity` از آن ارث بری شده است را بسته به نیاز خود بازنویسی<sup>۱</sup> کنیم تا کد مد نظر خود را در آن اجرا کنیم زمانیکه متدی را بازنویسی می‌کنیم دو گزینه پیش روی ما است:

۱. اول اینکه می‌توانیم کلیه خصوصیات اولیه متد مد نظر که توسط برنامه نویسان سیستم عامل اندروید ایجاد شده را بازنویسی کنیم و یا

۲. می‌توانیم طوری به بازنویسی یک متد پردازیم که علاوه بر خصوصیات اصلی‌اش، خصوصیات جدیدی را هم که برای آن در نظر می‌گیریم را در برگیرد که در این صورت می‌بایست از دستور `super` استفاده کنیم.

به عبارت دیگر زمانیکه از کلید واژه `super` در کد خود استفاده می‌کنیم، این دستور را به ماشین مجازی جاوا می‌دهیم تا علاوه بر کدهای موجود در کلاس مرتبط با متد `onCreate`، کدهای مد نظر ما را نیز اجرا نماید.

در حقیقت اگر کد `super.onCreate(savedInstanceState)` را از اپلیکیشن خود حذف کنیم این دستور را به ماشین مجازی می‌دهیم که فقط کد بازنویسی شده ما را اجرا کند و به هیچ وجه کدهای مرتبط با کلاس `Activity` که در برگرفته متد `onCreate` است را مد نظر قرار ندهد. اما نکته اینجا است که به هر حال این خط از کد برای اجرای اپلیکیشن الزامی است چرا که برای اجرای یک اپلیکیشن به طور کامل و بدون نقص به کدهای نوشته شده توسط برنامه نویسان سیستم عامل اندروید برای متد `onCreate` نیاز داریم. همانطور که در کد فوق مشاهده می‌شود داخل پرانتز مقابل متد `onCreate` پارامتر `Bundle` `savedInstanceState` قرار گرفته است. این پارامتر شی‌ای تحت عنوان `savedInstanceState` است که از روی کلاس `Bundle` ساخته شده است. وظیفه‌ای که این پارامتر بر عهده دارد این است که امکان ذخیره سازی مقادیر را به توسعه دهنده بدهد.

---

1- Override

در نهایت به متد `setContentView(R.layout.main)` می‌رسیم که این وظیفه را دارا است تا هرآنچه داخل آن قرار گرفت را به نمایش در آورد. در حقیقت با قرار دادن `R.layout.main` داخل پرانتز مقابل این متد این دستور را به ماشین مجازی می‌دهیم تا فایل `main.xml` را به نمایش در آورد.

#### ۴-۶-۱-۲ معرفی سرویس‌ها<sup>۱</sup>

دومین موافه‌ای<sup>۲</sup> که از میان دیگر مولفه‌ها از اهمیت بسزایی برخوردار است سرویس است. در حقیقت در پروسه طراحی اپلیکیشن اندروید سرویس به اجرای عملیاتی اطلاق می‌شود که خارج از دید کاربر رخ می‌دهند، ممکن است این دسته از عملیات خواه برای مدت زمان کوتاهی صورت پذیرند و خواه برای مدت زمان طولانی‌تری اجرا گردند. نکته‌ای که در مورد سرویس‌ها جالب است این است که آنها دارای هیچ گونه ظاهر گرافیکی نمی‌باشند و در پشت<sup>۳</sup> برنامه اجرا می‌شوند. برای روشن شدن مطلب مثالی ذکر می‌کنیم. فرض کنیم که در حین اس ام اس دادن به یکی از دوستان خود تمایل داریم تا به آهنگ مورد علاقه مان نیز گوش فرا دهیم. در حقیقت زمانیکه موسیقی در حال پخش شدن است و ما اقدام به اس ام اس دادن می‌کنیم ما هیچ اثری به جزء صدای آهنگ نخواهیم دید و این همان خاصیت سرویس‌ها است که گفته می‌شود در پس زمینه اپلیکیشن اتفاق می‌افتند. نکته دیگری که در مورد سرویس‌ها می‌بایست مد نظر قرار دهیم این است که سرویس‌ها مجزا از فعالیت‌هایی که آنها را اجرا می‌کنند می‌توانند به کار خود ادامه دهند.

#### ۴-۶-۱-۳ معرفی ارائه دهنده‌ی محتوا

جایگاه سوم در میان مولفه‌ها را ارائه دهنده محتوی به خود اختصاص داده‌است. وظیفه‌ای که بر عهده این گروه گذاشته شده‌است، ذخیره سازی اطلاعات و قرار دادن اطلاعات ذخیره شده در اختیار دیگر اپلیکیشن‌ها می‌باشد. به طور خلاصه تنها راه به اشتراک گذاری داده‌ها میان اپلیکیشن‌های مختلف به کارگیری ارائه دهنده‌ی محتوا می‌باشد. چنانچه ما داده‌هایی داشته باشیم و بخواهیم این داده‌ها را در

---

1- Services  
2- Component  
3- Background

دسترس دیگر بخش‌های اپلیکیشن خود قرار دهیم، به سادگی می‌توانیم یک ارائه دهنده محتوا ایجاد کرده و مابین بخش‌هایی که می‌خواهند از آن استفاده کنند ارتباطی برقرار سازیم. تعدادی از این ارائه دهنده‌های محتوا از پیش در سیستم عامل اندروید تعبیه شده‌اند که از آن جمله می‌توان به تماس<sup>۱</sup> و رسانه<sup>۲</sup> اشاره کرد. به طور مثال اگر بخواهیم در اپلیکیشنی به لیست شماره‌های تلفن همراه خود دسترسی پیدا کنیم، صرفاً نیاز است تا ارتباطی مابین اپلیکیشن خود و ارائه دهنده محتوا مرتبط با تماس سیستم عامل اندروید برقرار سازیم.

#### ۴-۱-۶-۴ معرفی گیرنده پخش

آخرین مولفه تحت عنوان گیرنده پخش این وظیفه را دارا است تا به هشدارهایی که در سطح کل سیستم بوجود می‌آیند پاسخ دهد. همانند سرویس‌ها گیرنده‌های پخش نیز دارای هیچ گونه ظاهرگرافیکی نمی‌باشند. برای روشن شدن وظیفه گیرنده پخش به مثالی اکتفا می‌کنیم، فرض کنید می‌خواهیم برنامه‌ای بنویسیم که زمانیکه انرژی باتری تلفن همراه ما به ۳۰ درصد رسید به ما هشدار دهد. این هشدار که به محض رسیدن انرژی تلفن همراه به ۳۰ درصد روی صفحه تلفن مشاهده می‌شود یک نوع گیرنده پخش می‌باشد. گیرنده‌ها می‌توانند به نوار وضعیت<sup>۳</sup>، اخطار<sup>۴</sup> بفرستند تا به کاربر هشدار بدهد.

#### ۴-۶-۲ معرفی فایل ظاهری

هر اپلیکیشن و یا بازی اندرویدی می‌بایست دارای فایل‌هایی تحت عنوان `AndroidManifest.xml` در دایرکتوری اصلی<sup>۵</sup> خود باشد. کاری که این فایل انجام می‌دهد این است که اطلاعاتی کلی پیرامون اپلیکیشن یا بازی طراحی شده به سیستم اندروید می‌دهد. به عبارت دیگر، این فایل موافه‌هایی را که ما در برنامه خود مورد استفاده قرار داده‌ایم را به سیستم عامل اندروید معرفی می‌کند. از سوی دیگر این

---

1- Contact  
2- Media  
3- Status Bar  
4- Notification  
5- Root Directory

فایل مجوزهایی<sup>۱</sup> که کاربر برای نصب برنامه می‌بایست صادر کند را نیز در برمی‌گیرد، فرض کنیم که برنامه‌ای طراحی کرده‌ایم که نیاز به دسترسی به اینترنت دارد، حال اگر کسی بخواهد این برنامه طراحی شده توسط ما را مورد استفاده قرار دهد، مجبور است در حین نصب اجازه دسترسی به اینترنت را به این اپلیکیشن صادر کند. همانطور که توضیح داده‌شد، سیستم عامل اندروید دارای نسخه‌های مختلفی است. چنانچه ما برنامه‌ای مثلا برای نسخه ۳,۰ طراحی کرده‌باشیم، این حداقل نسخه مورد نیاز برای اجرای اپلیکیشن در فایل Manifest ذکر خواهد شد و اگر کاربری که سیستم عامل اندروید گوشی‌اش پایین تر از نسخه ۳,۰ باشد بخواهد برنامه طراحی شده ما را استفاده کند در حین نصب فایل ظاهری این مسئله را خواهد فهمید و از نصب برنامه جلوگیری می‌شود. درواقع در کد موجود در این فایل دستور `android:minSdkVersion="7"` حداقل بسته توسعه جاوا مورد نیاز اپلیکیشن را مشخص کرده است. مواردی دیگری علاوه بر موارد فوق نیز در این فایل مشاهده می‌شود که در ذیل به آنها اشاره می‌کنیم: نامی که برای پکیج اپلیکیشن مورد استفاده قرار داده‌ایم در این فایل قرار گرفته‌است. علت قرار گرفتن نام پکیج در فایل ظاهری این است که نام این پکیج به منزله یک معرف منحصر به فرد از اپلیکیشن برای سیستم عامل اندروید می‌باشد.

همانطور که در کد زیر مشاهده می‌شود `package="com.behzadmoradi.mainpackage"` نشانگر نام پکیجی است که کلیه کلاس‌های جاوای اپلیکیشن در آن قرار گرفته‌اند. کلیه فعالیت‌ها و سرویس‌های به کار گرفته‌شده در اپلیکیشن که تحت عنوان مولفه‌های اپلیکیشن شناخته می‌شوند نیز در این فایل معرفی می‌گردند. در کد زیر خطی که حاوی `android:name=".MainActivity"` می‌باشد تنها فعالیتی که در حین ساخت پروژه تعریف شده‌است را نشان می‌دهد.

---

1- Permission



```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.untitled9"
    android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="10"/>
    <application android:label="@string/app_name">
        <activity android:name="MyActivity"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
    </application>
</manifest>
```

شکل ۴-۱۰ کدهای مربوط به فایل ظاهری

### ۳-۶-۴ معرفی پوشه‌های موجود در برنامه

#### ۱-۳-۶-۴ پوشه دارایی<sup>۱</sup>

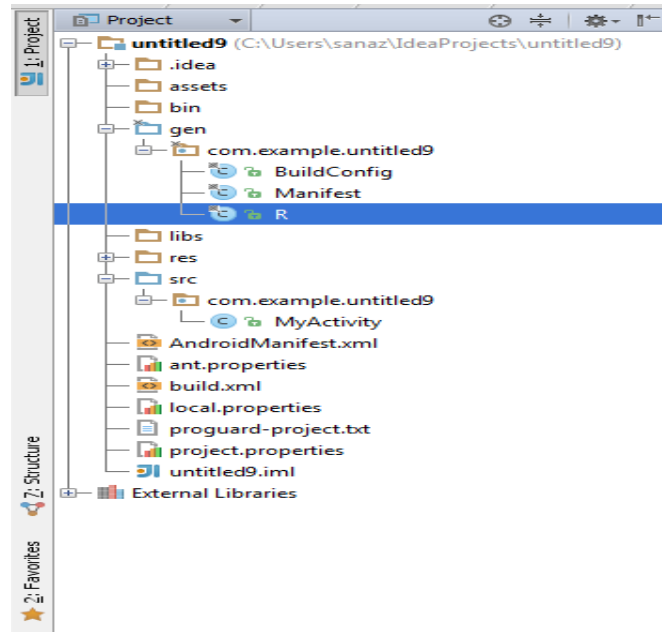
ما در دایرکتوری assets می‌توانیم تمامی فایل‌های صوتی، متنی و ... را به غیر از تصاویر قرار دهیم.

#### ۲-۳-۶-۴ پوشه bin

دایرکتوری bin که خروجی Build پروژه ما خواهد بود. در این دایرکتوری فایل‌ها با پسوند apk وجود دارد که همان فایل خروجی نرم افزار اندرویداست و قابل نصب بر روی هر گوشی اندرویدی خواهد بود.

#### ۴-۶-۳ پوشه gen

واژه gen مخفف واژه Generated به معنی "تولید شده" یا "حاصله" می‌باشد. این فولدر محل ذخیره سازی فایل جاوایی تحت عنوان R است.



شکل ۴-۱۱ نمایش پوشه حاصله

(حرف R از اول واژه Resource به معنی "منبع" گرفته شده است). به طور خلاصه می‌توان گفت که ما در توسعه اپلیکیشن اندرویدی از عناصر زیادی استفاده می‌کنیم که برخی از آنها توسط XML<sup>۱</sup> ایجاد می‌شوند مثل لایه‌ها، دکمه‌ها و غیره و برخی دیگر عناصری هستند همچون عکس‌ها، فایل‌های صوتی و ... که جزو منابع اپلیکیشن ما قرار دارند. حال برای آنکه بتوانیم به کلیه این عناصر درفعالیت‌ها دسترسی داشته باشیم بایستی آدرس آنها را داخل کلاس مرتبط با فعالیت وارد کنیم.

اگر بخواهیم مثلاً آدرس یک دکمه که در XML ساخته شده است را در کلاس خود بیاوریم، نه تنها این آدرس ممکن است طولانی شود، بلکه ممکن است در حین اجرا شدن اپلیکیشن نیز با مشکل مواجه شویم و در اینجا است که اهمیت فایل R دوچندان می‌شود. به عبارت دیگر سیستم عامل اندروید با این

1- Extensible Markup Language

کار جاوا و XML را به یکدیگر مرتبط می‌سازد. کاربرد این فایل به این شکل است که یک شناسه<sup>۱</sup> برای هر یک از عناصر موجود اپلیکیشن ما من جمله لایه‌ها، متون، عکس‌ها، دکمه‌ها و غیره در این فایل ساخته می‌شود. نکته جالب در ارتباط با این فایل این است که این فایل به صورت خودکار توسط برنامه ایجاد می‌گردد.

به عبارت دیگر به محض اینکه ما یک عنصر جدید مثلاً یک دکمه<sup>۲</sup> جدید در فایل ظاهری برنامه ایجاد می‌کنیم، اندروید به صورت خودکار به آن دکمه یک شناسه اختصاص داده و آن شناسه را وارد فایل R می‌کند. حال زمانی که بخواهیم به طول مثال از آن دکمه در یک فعالیت استفاده کنیم، از طریق شناسه آن دکمه در فایل R به آن دکمه دسترسی پیدا خواهیم کرد. به نظر می‌رسد علت انتخاب یک نام بسیار کوتاه تک حرفی تحت عنوان R برای این فایل این است که در حین آدرس دادن به عناصر موجود در اپلیکیشن زمان زیادی صرف نوشتن نام این فایل نکنیم (فقط با نوشتن R.id سپس نام عنصر مد نظر به آن دست پیدا خواهیم کرد).

#### ۴-۳-۶-۴ پوشه rec

پوشه res که منابع ما هستند شامل چندین پوشه است که اصلی ترین آن طرح<sup>۳</sup> می باشد در این پوشه فایل‌های XML برنامه ما قرار دارد، در واقع res شامل تمامی تصاویر، انیمیشن‌ها و به عبارتی منابعی است که در چگونگی اجرای برنامه اندروید نقش اساسی دارند. این دایرکتوری بسیار بسیار حساس بوده و در صورت اشتباهاتی که منجر از مطالعه نکردن صحیح منابع موجود در برنامه نویسی اندروید است به راحتی ما را دچار بحران می‌کند. پوشه دیگری که در res قرار دارد مقادیر<sup>۴</sup> است، شامل تمام ویژگی‌هایی است که در برنامه تعریف میکنیم در

---

1- id  
2- Button  
3- layout  
4- values

واقع بعداز تعریف و نام گذاری ارجاع میکنیم به آنها که به طور پیش فرض string.xml در این پوشه تعریف شده است.

#### ۴-۳-۵ پوشه src

در واقع واژه src مخفف واژه Source به معنی "منبع" می باشد. این فولدر در برگیرنده کلیه فایل های جاوا است که در اپلیکیشن خود مورد استفاده قرار می دهیم. زیر مجموعه این فولدر همان پکیجی قرار دارد که درحین ساخت پروژه تحت عنوان com.example.test ساخته می شود. در واقع زمانی که ما پروژه جدید به نام test ایجاد می کنیم این پکیج در src ایجاد میشود به طور کلی کاری که این پکیج می کند این است که از تداخل فایل های جاوا که برای توسعه اپلیکیشن خود طراحی می کنیم جلوگیری کرده و موجب می گردد کلاس های مورد استفاده سازماندهی شوند. [۶]

فصل پنجم

پیاده سازی

## ۱-۵ مقدمه

در این فصل مروری داریم بر نحوه‌ی پیاده‌سازی پروژه، همچنین روش‌های استفاده شده در پیاده‌سازی را بررسی خواهیم کرد.

## ۲-۵ پیاده‌سازی پایگاه داده

### ۱-۲-۵ معرفی جداول موجود در پایگاه داده

#### (۱) جدول کاربران

جدول کاربران شامل فیلدهای زیر است که شناسه کاربری و رمز عبور کاربران را در خود نگهداری می‌کند.

جدول ۱-۵ جدول کاربران

نام فیلد	نوع فیلد	Null
فیلد <b>ID</b>	number	Not null
فیلد <b>name</b>	Varchar2(20)	
فیلد <b>pass</b>	Varchar2(20)	

## ۲) جدول حسابها

این جدول شامل لیست حسابهای افراد و مشخصات حسابها می باشد.

جدول ۲-۵ جدول حسابها

نام فیلد	نوع فیلد	Null
فیلد ID	number	
فیلد accname	Varchar2(20)	
فیلد accnumber	number	Not null
فیلد balance	number	

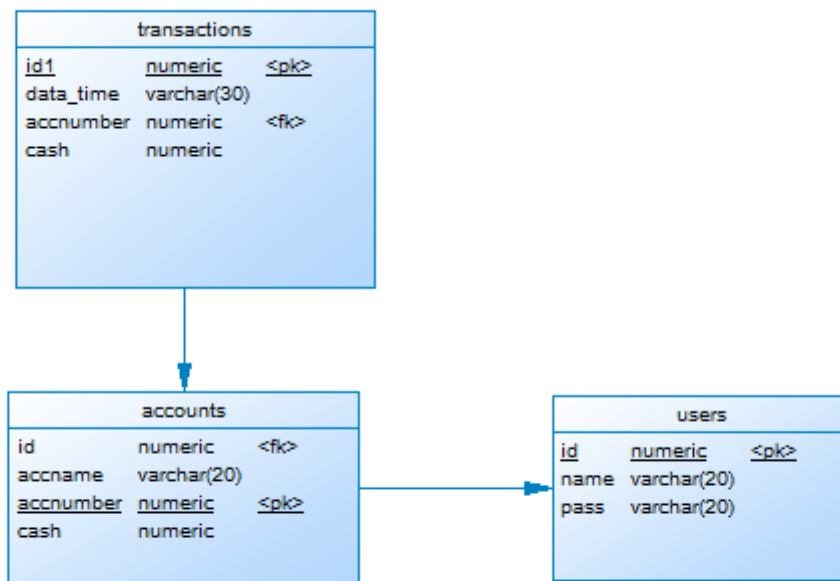
## ۳) جدول تراکنشها

اطلاعات مربوط به گردش حسابها در این جدول ذخیره می شود.

جدول ۳-۵ جدول تراکنشها

نام فیلد	نوع فیلد	Null
فیلد ID1	number	Not null
فیلد date_time	Varchar2(30)	
فیلد accnumber	number	
فیلد cash	number	

## ۵-۲-۲ ارتباط میان جداول پایگاه داده



شکل ۵-۱ نمایش ارتباط میان جداول پایگاه داده

## ۵-۳ معرفی معماری برنامه

برنامه نویسی سه لایه یا همان معماری MVC<sup>۱</sup> یکی از بهترین روشهای برنامه نویسی بوده و در این روش برنامه به سه لایه کلی تقسیم شده و هر لایه نیز از اجزای کوچکتری تشکیل می‌گردد. در این حالت کاربر که با فرمها در ارتباط می‌باشد به هیچ عنوان با ساختار بانک اطلاعات در ارتباط نبوده لایه‌های مختلف خدمات مورد نیاز را تامین می‌نماید.

### ۵-۳-۱ اجزای تشکیل دهنده معماری سه لایه

(۱) مدل: قسمتی از برنامه کاربردی است که مسئول بازیابی داده از بانک اطلاعاتی، ذخیره آن، تبدیل آن به شی یا آبجکتها و پیاده سازی منطق برنامه برای داده‌های مسئله است. در حقیقت بار اصلی معماری MVC بر عهده‌ی این بخش است



۲) **نمایشگر:** اجزایی از برنامه است که واسط کاربری برنامه را می‌سازد. معمولاً این واسط کاربری از داده-های مدل ساخته می‌شود. در واقع نقطه پایان برنامه کاربردی است. به کاربر نتایج عملیات و بازبازی و نمایش داده از طریق برقراری ارتباط با دو بخش دیگر، یعنی مدل و کنترلگر را نشان می‌دهد. برای مثال، هنگامی که کاربر در فرم ورود به سیستم رمز عبور خود را وارد می‌کند، اکثر برنامه نویسان در همان فرم اقدام به چک کردن رمز عبور می‌کنند که این عمل مغایر با قوانین MVC است. در MVC هنگامی که کاربر رمز عبور را وارد کرد، رمز عبور بدون هیچگونه اعمالی به بخش‌های دیگر فرستاده می‌شود و فقط یک نتیجه ساده یا خبر از بخش‌های دیگر دریافت می‌کند که از طریق آن اجازه‌ی ورود به برنامه داده می‌شود.

۳) **کنترلگر:** اجزایی از برنامه هستند که مدیریت تعامل با کاربر را بر عهده دارند. می‌توان گفت که واسط بین مدل و نمایشگر می‌باشند، یعنی با مدل کار می‌کنند و در انتها نمایشگری را برای نشان دادن واسط کاربری انتخاب می‌کنند. ورودی کاربر را مدیریت کرده و به آنها پاسخ می‌دهند و با کاربر تعامل می‌کنند. برای مثال، کنترلگر عبارت‌های پرس و جوی بانک اطلاعاتی را مدیریت کرده و آنها را به مدل ارسال می‌کند، وظیفه اجرای پرس و جویها با مدل است.

در این پروژه لایه نمایش در واقع فرم‌هایی هستند که در برنامه با اندروید پیاده سازی شده و به کاربر نشان داده می‌شود. لایه مدل نیز خود از سه بخش <sup>1</sup>BL, <sup>2</sup>DA, <sup>3</sup>TO در سمت سرور تشکیل شده است. در قسمت TO در واقع هر جدول از پایگاه داده به صورت یک کلاس تعریف شده و فیلدهای جداول، ویژگی‌های این کلاس‌ها می‌باشند هدف از این کار دسترسی راحت به اطلاعات و پیاده سازی مفهوم شی گزایی در سیستم می‌باشد

---

1- Business Logic  
2- Data Access  
3-Table Object

لایه DA همان لایه دسترسی داده‌است که وظیفه اعمال تغییرات در داده‌ها را بر عهده دارد و کارهایی چون ذخیره و بازیابی داده را انجام می‌دهند البته این امر نیازمند برقراری ارتباط با لایه‌ی TO می‌باشد.

لایه BL وظیفه دارد با لایه کنترلر و هر آنچه در قسمت کاربر وجود دارد ارتباط برقرار کند، بین لایه نمایش و لایه دسترسی داده قرار گرفته‌است تا کارها و درخواست‌هایی را که از سمت کاربر به سرور فرستاده می‌شود را پاسخ بدهد. [۷]

#### ۴-۵ نحوه‌ی برقراری ارتباط با پایگاه داده در جاوا

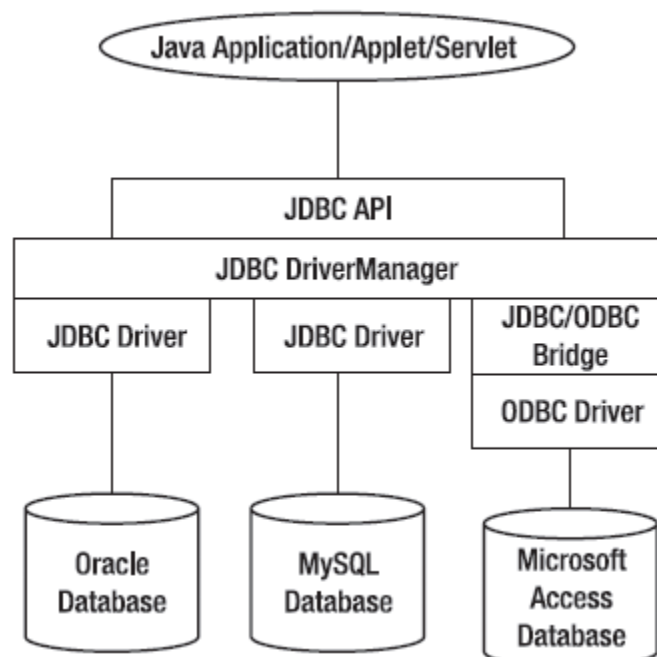
JDBC<sup>۱</sup> مجموعه‌ای از رابط‌های برنامه نویسی کاربردی<sup>۲</sup> است که اجازه اتصال آسان به یک دامنه وسیع از پایگاه داده‌ها (به خصوص پایگاه داده‌های رابطه‌ای) را می‌دهد.

در جاوا، رابط برنامه کاربردی JDBC به وسیله دو بسته تعریف شده است، `Java.sql` و `Javax.sql`. این بسته‌ها رابط برنامه نویسی کاربردی‌ای را برای دستیابی و پردازش داده‌های ذخیره شده در یک منبع داده با استفاده از زبان جاوا فراهم می‌کند. در واقع اصلی‌ترین و متداول‌ترین اشیاء مورد استفاده مانند (`connection,ResultSet,Statement,PreparedStatement`) و همچنین کلاس‌ها و میانجی‌هایی برای به دست آوردن پایگاه داده را ایجاد می‌کند.

شکل زیر نشان می‌دهد که چگونه یک برنامه از JDBC برای تعامل با پایگاه داده استفاده می‌کند.

---

1- Java Data Base Connectivity  
2- application programming interface(API)



شکل ۵-۲ نحوه برقراری ارتباط با پایگاه داده (مقاله پایگاه داده، حسن امیدی)

مدیر درایور<sup>۱</sup>، درایورهای JDBC را در حافظه بارگذاری می کند و همچنین می تواند برای ساختن ارتباط و شی Connection استفاده شود. در واقع برای برقراری ارتباط بین برنامه جاوا و پایگاه داده به پل ارتباطی نیازمندیم که ساختن این پل توسط مدیر درایور صورت می گیرد با استفاده از شی Connection. بعد از ساخته شدن ارتباط حال باید دستورات SQL را به سمت پایگاه داده هدایت کنیم در واقع دستوراتی نظیر (درج، حذف، به روزرسانی) باید اجرا شوند که توسط Statement این کار انجام میشود. نتیجه ی درخواستها با Resultset نمایش داده می شود. [۸]

---

1- DriverManager

## ۵-۵ نحوه ارتباط بین سرور و کاربر

### ۵-۵-۱ مفهوم REST<sup>۱</sup>

در واقع یک معماری جدید در وب سرویس می‌باشد که از پروتکل HTTP<sup>۲</sup> برای ارتباط بین دو سیستم (client-server) استفاده می‌کند.

### ۵-۵-۲ مفهوم Websocket

یک پروتکل است که کانال‌های ارتباطی دوطرفه را مهیا می‌کند و باعث تعامل بین سرور و کاربر می‌شود.

### ۵-۵-۳ مفهوم Json<sup>۳</sup>

یک استاندارد متنی برای انتقال داده‌ها می‌باشد. به کمک این شیوه اشیاء و اطلاعات برنامه‌نویسی به متن قابل پردازش توسط رایانه تبدیل می‌شود، نکته‌ی مهم در این شیوه قابلیت فهم آسان توسط انسان است.

تمام زبان‌های برنامه‌نویسی متداول با کمک کتابخانه‌های جانبی از JSON پشتیبانی می‌کنند. [۹]

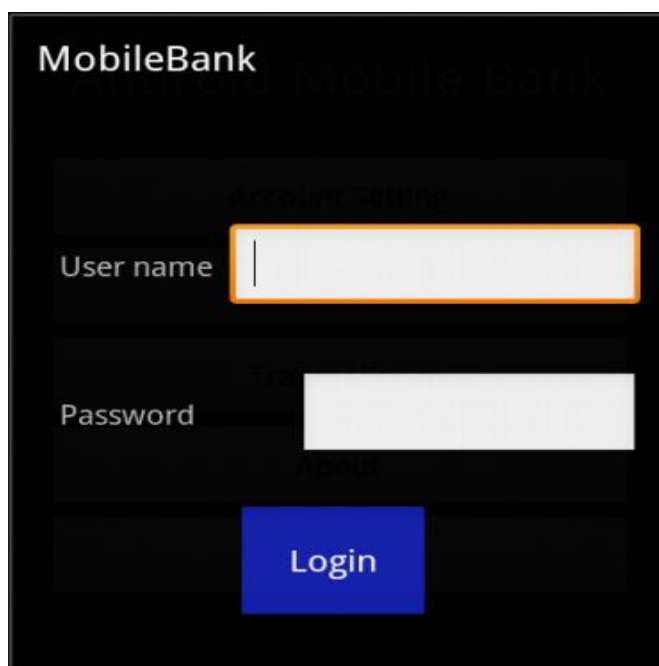
---

1- Representational State Transfer  
2- Hypertext Transfer Protocol  
3- JavaScript Object Notation

## ۵-۶ معرفی فعالیتها و صفحات موجود در برنامه

### ۵-۶-۱ صفحه مربوط به ورود

در این صفحه کاربر برای ورود به برنامه، شناسه و رمز خود را وارد می‌کند. در صورت صحت اطلاعات وارد شده کاربر وارد منو اصلی می‌شود. اگر اطلاعات صحیح نباشد به کاربر پیغام خطا نمایش داده می‌شود و همچنین اگر سرور قطع باشد پیغام قطع بودن ارتباط نمایش داده می‌شود.



MobileBank

User name

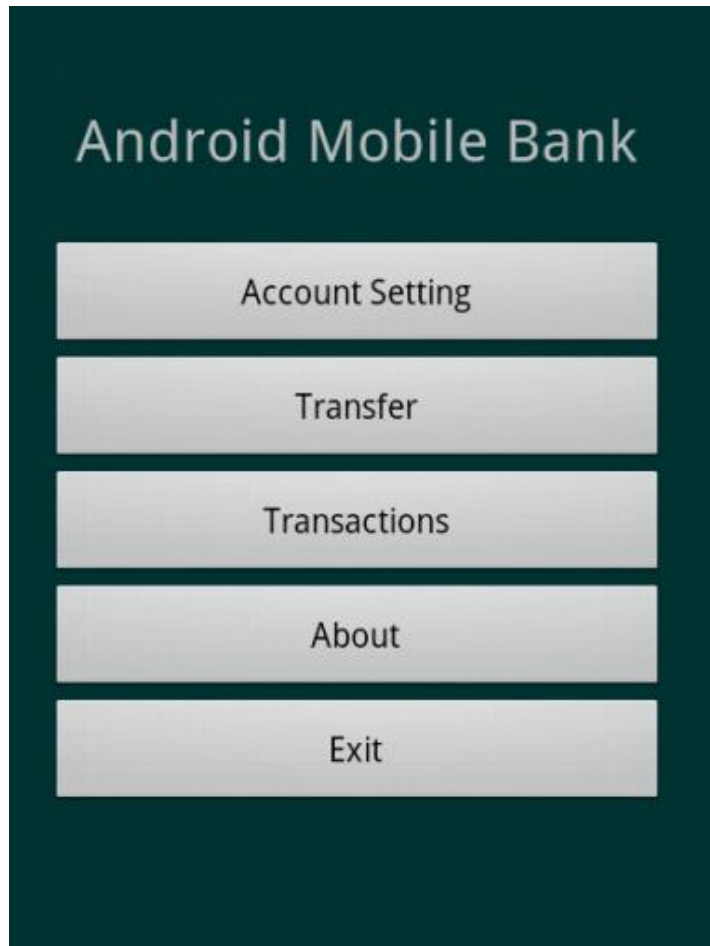
Password

Login

شکل ۵-۳ صفحه ورود

## ۵-۶-۲ صفحه مربوط به منو اصلی

کاربر بعد از ورود باید یکی از عملیات مد نظر خود را از منو اصلی انتخاب کند. در این منو گزینه‌های تنظیمات حساب، انتقال وجه، مشاهده تراکنش‌ها، درباره ما، تغییر رمز و خروج مشاهده می‌شود.



شکل ۵-۴ صفحه منو اصلی

### ۵-۶-۳ صفحه مربوط به تنظیمات حساب

در این صفحه کاربر مشخصات حساب‌های خود را مشاهده می‌کند. در صورت نیاز می‌تواند حساب جدیدی اضافه کند و یا حسابی را حذف کند. همچنین با کلیک بر روی هر کدام از شماره حساب‌ها می‌تواند اطلاعات حساب را به فرد خاصی ارسال کند.

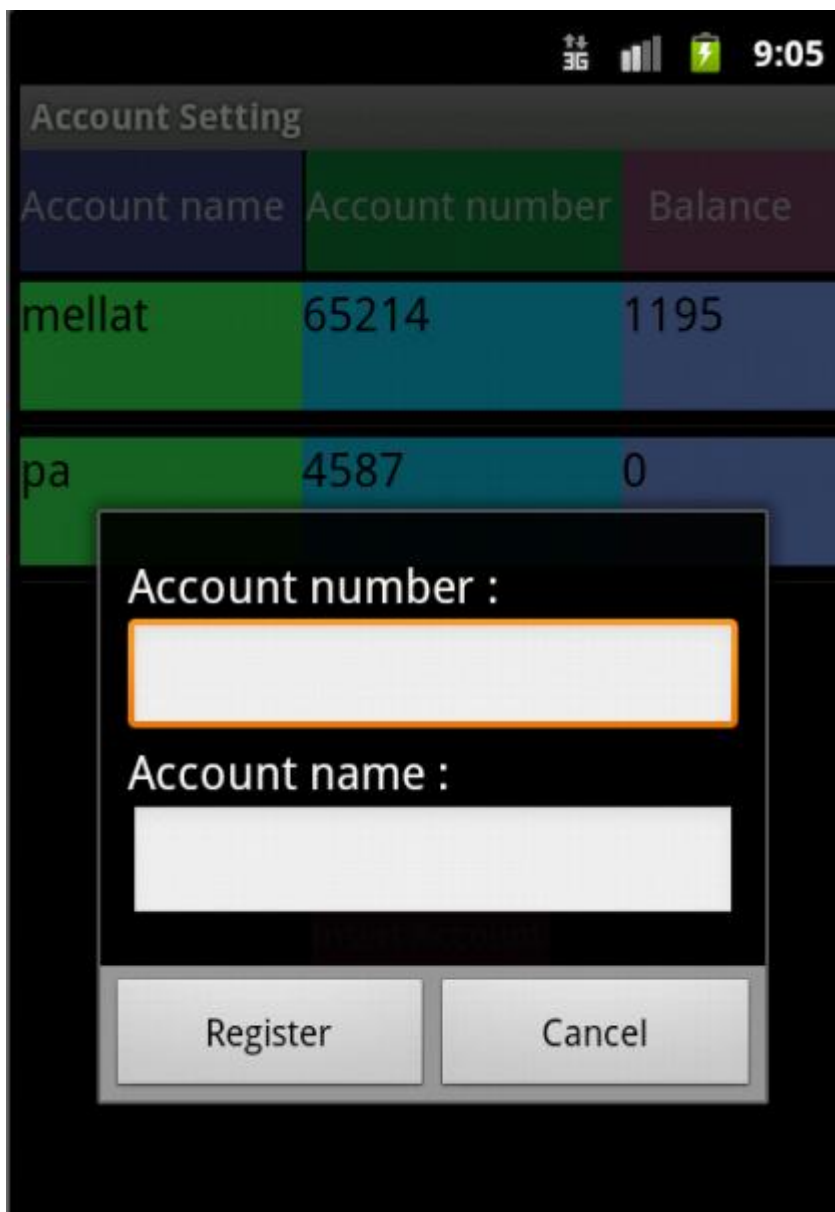
Account Setting		
Account name	Account number	Balance
mellat	65214	1195
pa	4587	0

[Insert Account](#)

شکل ۵-۵ صفحه تنظیمات حساب

### ۵-۶-۳-۱ صفحه مربوط به اضافه کردن حساب جدید

کاربر نام و شماره حساب را وارد کرده و اطلاعات را ثبت می کند و در صورت ثبت شدن اطلاعات پیغامی به کاربر نمایش داده می شود.

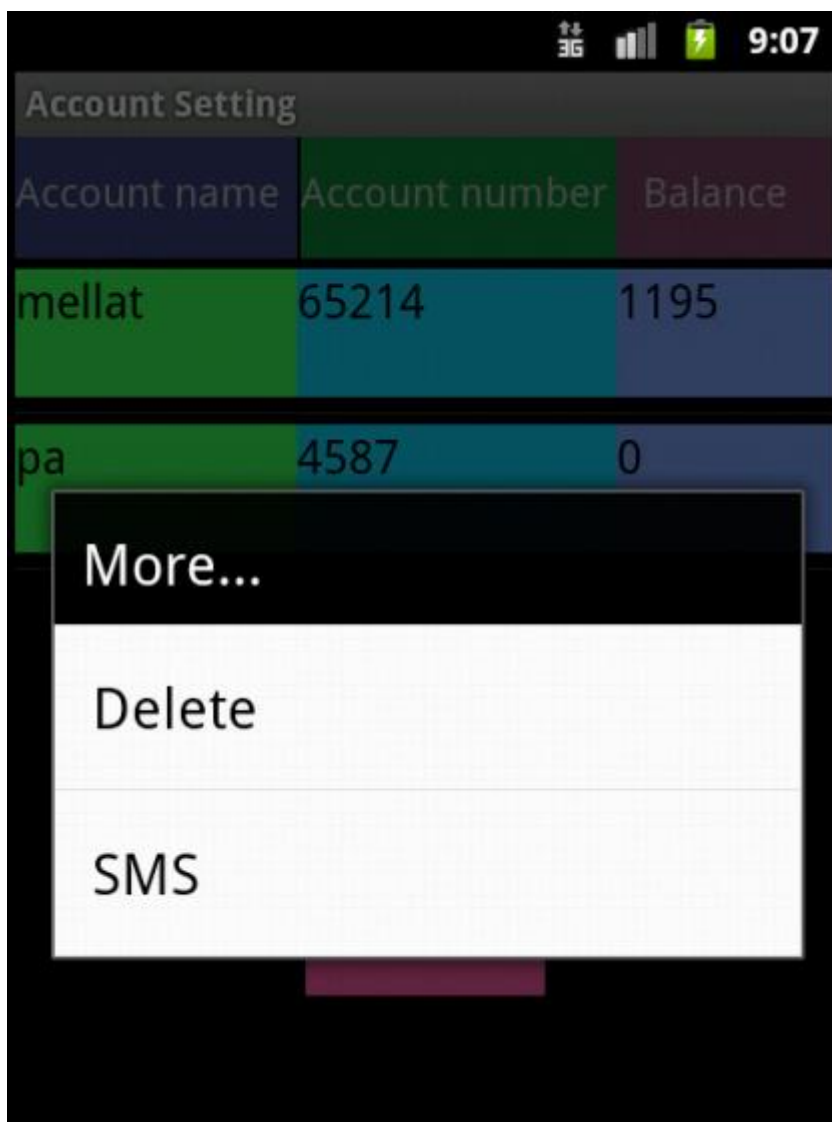


شکل ۵-۶ صفحه اضافه کردن حساب جدید



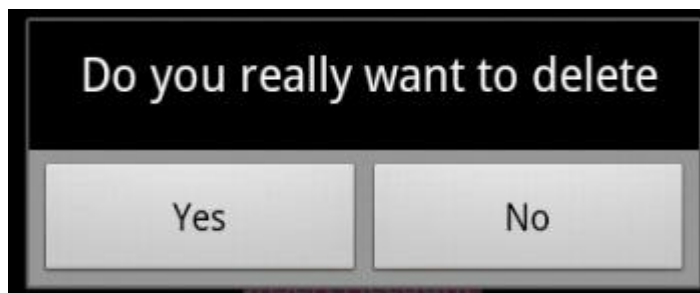
## ۵-۶-۳-۲ صفحه مربوط به حذف حساب و ارسال پیام

کاربر با انتخاب هر کدام از حساب‌ها، صفحه‌ی زیر را مشاهده می‌کند که می‌تواند آن حساب را حذف و یا شماره حساب را به فرد مورد نظر ارسال کند



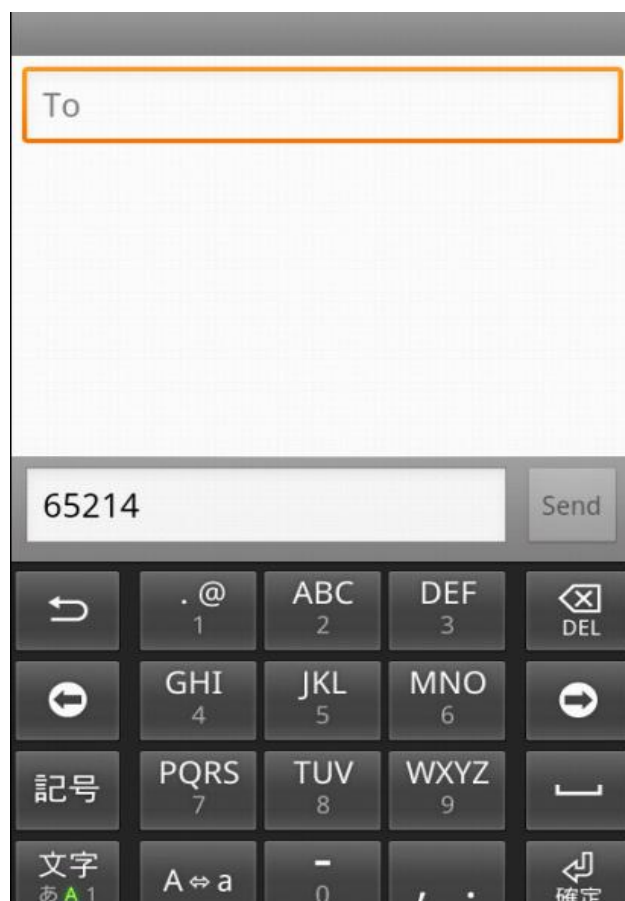
شکل ۵-۷ صفحه حذف حساب و ارسال پیام

هنگام حذف از کاربر تأیید حذف حساب درخواست می‌شود



شکل ۵-۸ صفحه تأیید حذف حساب


اگر کاربر گزینه پیام را انتخاب کند صفحه زیر نمایش داده می‌شود و کاربر باید شماره تلفن گیرنده را وارد کرده و پیام را ارسال کند. در صورت ارسال پیام، دوباره کاربر به صفحه‌ی تنظیمات حساب برمی‌گردد.



شکل ۵-۹ صفحه ارسال پیام

۴-۶-۵ صفحه مربوط به انتقال وجه

در این صفحه کاربر شماره حساب خود را انتخاب کرده، شماره حساب مقصد و مبلغ را وارد می‌کند و دکمه next را می‌زند.



The screenshot shows a mobile application interface for a transfer. At the top, the status bar displays '3G', signal strength, battery, and the time '9:28'. The main screen has a dark background with white text. The title 'Transfer' is at the top. Below it, 'Source Account :' is followed by a dropdown menu showing 'mellat 65214'. Underneath is 'Destination Account :', followed by a redacted blue bar. Below that is 'Cash :', followed by another redacted blue bar. At the bottom, there is a large white button with the text 'Next'.

شکل ۱۰-۵ صفحه انتقال وجه

در ادامه، صفحه‌ی گرفتن تائید به کاربر نمایش داده می‌شود و کاربر با فشردن دکمه Register اطلاعات وارد کرده را تائید و درخواست خود را به سمت سرور می‌فرستد. بعد از تائید موجودی حساب بررسی شده اگر موجودی کافی نباشد و همچنین اطلاعات مقصد صحیح نباشد پیغام خطا به کاربر نمایش داده میشود.

Source Account :	
65214	
Destination Account :	
1234	
Cash :	
50	
Register	Return

شکل ۵-۱۱ صفحه گرفتن تائید

## ۵-۶-۵ صفحه نمایش تراکنش‌ها

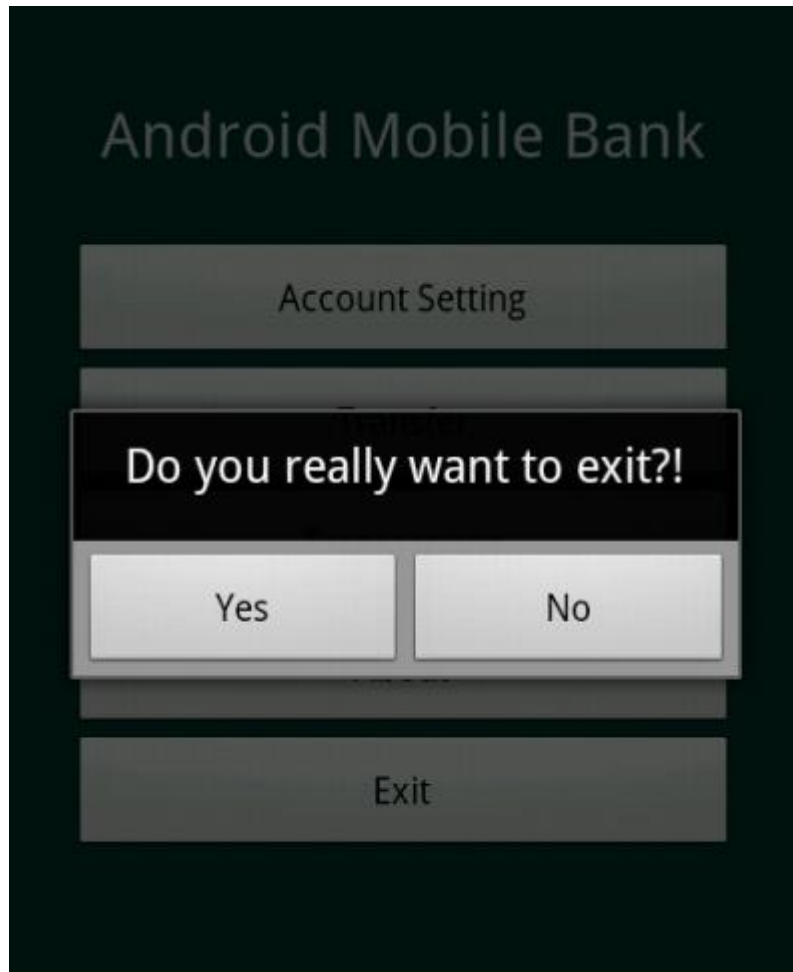
در این صفحه کاربر ابتدا شماره حساب خود را انتخاب کرده سپس برای مشاهده لیست تراکنش‌ها دکمه نمایش را می‌زند. در این لیست اطلاعات پنج گردش آخر حساب وجود دارد. که شامل تاریخ انجام تراکنش و نوع تراکنش که واریزی یا برداشتی است، می‌باشد.

Cash	Date
500	Tue Apr 08 12:37:21 IRDT 2014
695	Tue Apr 08 11:24:53 IRDT 2014
-50	Fri May 02 12:58:50

شکل ۵-۱۲ صفحه نمایش تراکنش‌ها

## ۵-۶-۶ صفحه خروج

کاربر با انتخاب گزینه‌ی خروج از منو اصلی صفحه زیر را مشاهده می‌کند سپس در صورت اطمینان از خارج شدن از برنامه دکمه yes را زده و برنامه بسته می‌شود.



شکل ۵-۱۳ صفحه خروج از برنامه

## ۵-۶-۷ صفحه مربوط به تغییر رمز عبور

در این صفحه کاربر برای تغییر رمز عبور، ابتدا رمز فعلی خود را وارد کرده و سپس رمز جدید را تعریف و دوبار وارد می‌کند و دکمه ok را می‌زند. کاربر اگر رمز فعلی را وارد نکرده باشد و یا رمز جدید خود را که دوبار وارد می‌کند یکسان نباشد پیغام خطا نشان داده می‌شود.

The image shows a 'Change Password' dialog box. It has a dark background and a title bar at the top that says 'Change Password'. Below the title bar, there are three input fields: 'Current Password', 'New Password', and 'Confirm New Password'. The 'Current Password' field is highlighted with an orange border. At the bottom center, there is an 'OK' button.

شکل ۵-۱۴ صفحه تغییر رمز عبور

## فصل ششم

### جمع‌بندی و نتیجه‌گیری



## ۶-۱ نتیجه‌گیری

هدف نهایی این پژوهش طراحی برنامه موبایل بانک بود که این امر پس از بررسی و تحقیق بر روی نیازها و همچنین مزایای وجود این برنامه در بهبود اثر بخشی و کارایی عملیات بانکی، محقق گردید. این تحقیق موفق شد به یک مدل جامع دست یابد؛ و آن را در قالب یک وب سرویس و برنامه‌ی موبایل بانک طراحی و با سیستم عامل اندروید به کاربران خود خدمات بانکی ارائه کند.

می‌توان نتیجه گرفت که با اجرا و استفاده از این برنامه:

- از فضای کوچک فیزیکی پاتراتر گذاشته و جهانی می‌شویم
  - مشتری به جای حضور فیزیکی در بانک و پرداخت هزینه‌های رفت‌وآمد با داشتن این برنامه می‌تواند عملیات بانکی خود را در هر ساعت از شبانه روز انجام دهد و این منجر به صرف‌جویی در هزینه و زمان می‌شود.
  - با کاهش کارهای زائد اداری هزینه‌ها تا حد قابل توجهی کاهش می‌یابد.
  - کاهش نیروی انسانی موجب صرفه‌جویی در هزینه‌های بانک نیز می‌شود.
- گسترش فرهنگ استفاده از اینترنت و تکنولوژی‌های روز چون موبایل بانک، گامی در جهت رسیدن به اهداف بانکداری الکترونیک و تجارت الکترونیک می‌باشد.

## ۶-۲ پیشنهادات

در این قسمت، به منظور ارائه راهکاری در جهت توسعه برنامه پیاده‌سازی شده، به بیان پیشنهاداتی در زمینه‌های مختلف کاری برنامه اقدام می‌شود. این پیشنهادات به صورت ذیل تشریح می‌گردند:

۱. توسعه دامنه فعالیت برنامه موبایل بانک. در این زمینه به نظر می‌رسد که با افزایش دامنه کاربرد این برنامه در زمینه‌هایی مثل خرید شارژ تلفن، پرداخت قبوض و استعلام چک می‌توان به مقبولیت بیشتری نزد کاربران دست یافت.
  ۲. ایجاد پایگاه داده‌ای پیچیده‌تر. بدیهی است که با توجه به موارد گفته شده در بند قبل، پایگاه داده ای متناسب با نیازهای جدید بایستی تهیه گردد. این پایگاه امکانات جدید برنامه را پشتیبانی خواهد کرد و از این رو حوزه اطلاعاتی گسترش خواهد یافت.
- در پایان، امید است پروژه ارائه شده مورد استفاده دانشجویان و محققان قرار گرفته‌باشد.



## منابع

[۱] فصل نامه علمی کاربردی بانک مسکن (پویش)، سال دوم، شماره ششم، تابستان ۹۲

[۲] بانکداری تلفن همراه. قابل دسترس در:

[http://en.wikipedia.org/wiki/Mobile\\_banking](http://en.wikipedia.org/wiki/Mobile_banking)

[۳] مروری بر RUP و قابلیت‌های آن در تولید نرم‌افزار (۱۳۸۴). قابل دسترس در:

[www.mgtsolution.com/olib/462284035.aspx](http://www.mgtsolution.com/olib/462284035.aspx)

[۴] آموزش UML سری شومز، حامد شایان - مهرداد سلامی، انتشارات فکر خلاق (۱۳۸۸)

[۵] تاریخچه سیستم‌عامل اندروید. قابل دسترس در:

<http://www.zoomit.ir/it-news>

[۶] آموزش برنامه‌نویسی اندروید. قابل دسترس در:

<http://androidcode.ir/>

[۷] معماری سه لایه. قابل دسترس در:

<http://fa.wikipedia.org/wiki/>

[۸] مقاله‌ی نحوه اتصال پایگاه‌داده و جاوا، حسن امیدی (۱۳۹۱). قابل دسترس در:

<http://irandoc.ir/>