دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فن‌آوری اطلاعات

پایان‌نامه کارشناسی

گرایش نرم‌افزار

عنوان

طراحی و پیاده سازی ابزاری با قابلیت توسعه‌پذیری به منظور استخراج داده‌های پاک‌شده از گوشی‌های هوشمند

نگارش

احسان عدالت

استاد راهنما

جناب آقای دکتر بابک صادقیان

شهریور ۱۳۹۴

به نام خدا

تعهدنامه اصالت اثر

تاریخ:

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

اینجانب احسان عدالت متعهد می‌شوم که مطالب مندرج در این پایان نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

احسان عدالت

امضا

با سپاس از سه وجود مقدس:

آنان که ناتوان شدند تا ما به توانایی برسیم...

موهایشان سپید شد تا ما روسفید شویم...

و عاشقانه سوختند تا گرمابخش وجود ما و روشنگر راهمان باشند...

پدرانمان

مادرانمان

استادانمان

تقدیر و تشکر:

سپاس و ستایش مر خدای را جل و جلاله که آثار قدرت او بر چهره روز روشن، تابان است و انوار حکمت او در دل شب تار، درفشان. آفریدگاری که خویشتن را به ما شناساند و درهای علم را بر ما گشود و عمری و فرصتی عطا فرمود تا بدان، بنده ضعیف خویش را در طریق علم و معرفت بیازماید.

بدون شک جایگاه و منزلت معلم، بالاتر از آن است که در مقام قدردانی از زحمات بی شائبه‌ی او، با زبان قاصر و دست ناتوان، چیزی بنگارم. اما از آنجا که تجلیل از معلم، سپاس از انسانی است که هدف آفرینش را تامین می‌کند، به رسم ادب دست به قلم برده‌ام، باشد که این خردترین بخشی از زحمات آنان را سپاس گوید.

از پدر و مادر مهربانم، این دو معلم بزرگوار که همواره بر کوتاهی من، قلم عفو کشیده و کریمانه از کنار غفلت‌های گذشته‌اند و در تمام عرصه‌های زندگی یار و یاورم بوده‌اند؛

از استاد با کمالات، جناب آقای دکتر بابک صادقیان که در کمال سعه صدر، با حسن خلق و فروتنی، از هیچ کمکی در این عرصه بر من دریغ نداشتند؛

از اساتید محترم، جناب آقای دکتر مهران سلیمان فلاح و آقای دکتر بهمن پوروطن که زحمت داوری این رساله را متقبل شدند؛

و در پایان، از حمایت‌ها و دلسوزی‌های دوستان عزیزم، آقایان سید محمد مهدی احمدپناه، حمیدرضا رمضانی و احمد اسدی که در طول پروژه از راهنمایی‌هایشان استفاده کردم؛

کمال تشکر و قدردانی را دارم.

چکیده

تلفن‌های همراه هوشمند به ابزارهایی اتلاق می‌شود که قادر بـه اجـرای قابلیـت‌هـای رایانـه هسـتند کـه معمولا دارای صفحه لمسی، قابلیت اتصال به اینترنت و سیستم‌عامل هستند که وظایف مربوطه را کنتـرل می‌کند. به دلیل آسانی استفاده و قیمت مناسب مردم روز به روز تمایل بیشتری به استفاده از این نـوع از ابزارها پیدا کرده‌اند. این ابزارها دارای قابلیت‌هـای گونـاگونی از جملـه ارسـال و دریافـت پیـام از طریـق سرویس پیام کوتاه یا ایمیل، نگهداری عکس، یافتن موقعیت جغرافیایی، مرور اینترنت و ... هستند که هر کدام از این قابلیت‌ها در قالب یک نرم‌افزار پیاده‌سازی شده‌اند. هر کدام از این نرم‌افزارها اطلاعات زیـادی در مورد دارنده گوشی در خود ذخیره کرده‌اند.

امروزه یکی از راه‌هایی که مراجع قضایی و پلیس از آنها برای استخراج اطلاعـات بـه منظـور اثبـات جـرم استفاده می‌کنند، اطلاعات ذخیره شده در گوشی‌های تلفن هوشمند است. این موضوع اهمیـت داده‌هـای ذخیره‌شده را نشان می دهد. در این پروژه استخراج این داده‌ها و همچنـین روش‌هـای بازیـابی داده‌هـای پاک‌شده بررسی شده است.

یکـی از روش‌هـایی کـه بـه واسـطه آن نـرم‌افزارهـا داده‌هـای خـود را ذخیـره مـی‌کننـد، اسـتفاده از پایگاه‌داده‌هاست. پایگاه‌داده Sqlite به دلیل سرعت و حجم حافظه اشغالی کم برای ایـن منظـور اسـتفاده می‌شود. ابزار پیاده‌سازی شده در این پروژه دارای این ویژگی است که پایگاه‌داده مورد اسـتفاده نـرم‌افـزار انتخابی، از گوشی تلفن کپی شده و داده‌هـای آن بـه تفکیـک جـداول موجـود در پایگـاه‌داده، اسـتخراج می‌شود. همچنین داده‌های پاک‌شده به سه روش استخراج مـی شـوند کـه عبارتنـد از بازیـابی از فضـای بلااستفاده، بازیابی از بلوک‌های آزاد و بازیابی به کمک فایل ژورنال. همچنین این نرم‌افزار می‌تواند داده‌هـا را بر اساس وجود رشته‌ای خاص بازیابی کند.

سیستم‌عامل‌های محبوب میان کاربران اندروید، iOS و ویندوز هستند که برای هـر کـدام روشـی خـاص برای استخراج فایل پایگاه‌داده نیاز است. نرم‌افزار پیاده‌سازی شده دارای قابلیت توسعه‌پذیری است که بـه واسطه آن می‌توان روش‌های مختلف اتصال و دریافت فایل پایگاه‌داده از سیسـتم‌عامـل‌هـا و پلتفـرم‌هـای مختلف را پیاده سازی و به سیستم اضافه کرد. در ایـن پـروژه، روش مـورد اسـتفاده بـرای دریافـت فایـل پایگاه‌داده از سیستم‌عامل اندروید ارائه شده‌است.

واژه‌های کلیدی:

پایگاه‌داده Sqlite، استخراج داده‌های پاک‌شده، توسعه‌پذیری، سیستم‌عامل اندروید

فهرست عناوین

صفحه

فهرست اشکال

صفحه

فهرست جداول

صفحه

۱

فصل اول

مقدمه

امروزه با پیشرفت تکنولوژی ابزارهای دیجیتالی گوناگونی از قبیل رایانه‌های همراه، تلفن‌های همراه و تبلت‌ها به صورت گسترده در اختیار عموم مردم است. این ابزارها در سال‌های اخیر به دلیل قیمت مناسب و همچنین آسانی فراگیری و استفاده از آنها در میان مردم محبوبیت پیدا کرده‌اند. در میان این ابزارها، گوشی‌های هوشمند در میان مردم جایگاه ویژه‌ای دارند و استفاده از آنها امری رایج و اجتناب ناپذیر است.

گوشی‌های هوشمند بدلیل قابلیت اتصال به اینترنت و همچنین پشتیبانی از ذخیره‌سازی اطلاعات، تبدیل به یکی از مراجعی شده‌اند که پلیس برای استخراج اطلاعات به منظور انجام تحقیقات و اثبات جرم از آنها استفاده می‌کند. این اطلاعات شامل عکس‌های ذخیره شده، مکان‌های مراجعه شده، سایت‌های بازدید شده، پیام‌های ارسال شده از طریق رایانامه یا سیستم پیام کوتاه یا نرم‌افزارهای پیام‌رسان از قبیل لاین[۱]، فهرست اسامی و شماره تلفن یا رایانامه اشخاص و هزاران نرم‌افزار دیگر که از طریق آنها امکان ذخیره‌سازی، ارسال یا دریافت اطلاعات در قالب‌های گوناگون (متن، عکس و ...) وجود دارد.

گوناگونی و حجم زیاد این اطلاعات اهمیت استخراج این اطلاعات را دوچندان می‌کند. علاوه بر اطلاعات حاضر، اطلاعات زیادی ممکن است به صورت سهوی یا عمدی پاک شوند که اهمیت آنها، لزوم بازیابی آنها را تایید می‌کند.

[۱] Line

۱.۱ نمای کلی پروژه

در این پروژه روش استخراج و بازیابی اطلاعات ذخیره شده در نرم‌افزارهایی که از پایگاه‌داده Sqlite پشتیبانی می‌کنند، بیان خواهد شد. لازم به ذکر است که نرم‌افزارهای گوشی‌های هوشمند مبتنی بر سیستم‌عامل اندروید و iOS و همچنین برخی از نرم‌افزارهای مبتنی بر سیستم‌عامل ویندوز برای ذخیره‌سازی اطلاعات خود از پایگاه‌داده Sqlite استفاده می‌کنند. در این پروژه روش استخراج پایگاه‌داده از گوشی‌های تلفن اندرویدی بررسی می‌شود و سپس اطلاعات موجود در آنها استخراج و بازیابی خواهند شد.

هسته اصلی سیستم، کتابخانه استخراج و بازیابی اطلاعات پایگاه‌داده است که عملکرد و الگوریتم‌های آن به تفصیل بیان خواهند شد. همان طور که ذکر شد پایگاه‌داده Sqlite در سیستم‌عامل‌های گوناگون کاربرد دارد. برای پشتیبانی از پلتفرم‌های مختلف لازم است ابزار پیاده‌سازی شده از قابلیت توسعه‌پذیری پشتیبانی کند. نحوه پیاده‌سازی و اجرای این قابلیت نیز در ادامه بیان خواهد شد. برای استخراج پایگاه‌داده‌ها از گوشی های اندرویدی از یک کتابخانه استفاده خواهد شد که به تلفن متصل شده و فایل‌های لازم را از آن استخراج می‌کند.

۲.۱ ساختار پایان نامه

در ادامه‌ی پایان‌نامه در فصل دوم به طور مختصر گوشی‌های تلفن همراه اندرویدی مورد بررسی قرار می‌گیرند. در فصل سوم پایگاه‌داده Sqlite و ساختار آن مورد بررسی قرار می گیرد. در فصل چهارم روش‌ها و الگوریتم‌های بازیابی اطلاعات پاک‌شده و تغییریافته بیان می‌شود. در فصل پنجم نحوه پیاده‌سازی نرم‌افزار و تحلیل و طراحی آن و اجرای قابلیت توسعه‌پذیری بررسی می‌شوند و سرانجام در فصل ششم به جمع‌بندی و کارهای آینده پرداخته خواهد شد.

۲

فصل دوم

بررسی اجمالی گوشی‌های تلفن هوشمند مبتنی بر سیستم‌عامل اندروید

اندروید در یونانی به معنای مرد، انسان، شبه آدم یا ربات نام سیستم‌عاملی است که گوگل برای تلفن‌های هوشمند و تبلت‌ها و هم اکنون برای تلویزیون‌ها عرضه می‌نماید و با همکاری ده‌ها شرکت بر روی دستگاه‌های مبتنی بر اندروید قرار می‌دهد. اندروید بر پایه هسته لینوکس ساخته شده است.

شکل ۱ لوگوی اندروید

در اوت ۲۰۰۵، گوگل شرکت اندروید واقع در پالو آلتو، کالیفرنیا را خرید. شرکت کوچک اندروید که توسط اندی رابین، ریچ ماینرز، نیک سیرز و کریس وایت پایه‌گذاری شده بود، در زمینه تولید نرم‌افزار و برنامه‌های کاربردی برای تلفن‌های همراه فعالیت می‌کرد. اندی رابین مدیر عامل اجرایی این شرکت پس از پیوستن اندروید به گوگل به سمت قائم‌مقام مدیریت مهندسی این شرکت و مسئول پروژه اندروید در گوگل منصوب شد.

شکل ۲ صفحه خانگی اندروید نسخه ۵,۰

نخستین گوشی مبتنی بر اندروید توسط شرکت اچ‌تی‌سی با همکاری تی-موبایل تولید شد. این گوشی به فاصله کمتر از یک سال از تشکیل اتحادیه گوشی باز یعنی در ۲۲ اکتبر ۲۰۰۸ تولید شد. در ۳ سپتامبر ۲۰۱۳ توسعه‌دهندگان اندروید به‌طور رسمی اعلام کردند که با شرکت نستله، که از شرکت‌های مطرح صنعت شکلات‌سازی جهان می‌باشد، همکاری خواهند کرد. در همین راستا نگارش ۴,۴ سیستم‌عامل اندروید، کیت‌کت نام گرفت .کیت کت از مارک‌های معروف شکلات است که توسط شرکت نستله

تولید می‌شود. آخرین نسخه اندروید، یعنی نسخه ۵,۰، نیز ۵ جولای ۲۰۱۴ با نام آب‌نبات چوبی عرضه شده است.

نرم‌افزارهای جانبی اندرویدی با استفاده از زبان جاوا نوشته می‌شوند و برای ارتباط با لایه‌های زیرین سیستم عامل می‌توانند از کتابخانه‌های جاوایی اندروید استفاده کنند. بخش رابط کاربری سیستم عامل اندروید با زبان جاوا نوشته شده‌است و بسیاری از برنامه‌های اندروید هم با جاوا نوشته شده‌اند. اما این سیستم عامل، Java Virtual Machine ندارد. برای اجرای برنامه‌های جاوایی روی این سیستم عامل، کدهای جاوا به کدهای Dalvik تبدیل می‌شوند و سپس روی Dalvik vitrual machine اجرا می‌شوند. دالویک یک ماشین مجازی جاوایی است که برای سیستم عامل اندروید بهینه شده‌است تا هم RAM و هم CPU و هم باتری کمتری مصرف کند. برنامه‌های جاوایی معمولی هم که روی گوشی‌های دیگر اجرا می‌شوند با استفاده از نرم‌افزارهای شبیه‌ساز ماشین مجازی جاوا مانند j2ME MIDP Runner روی این سیستم عامل قابل اجرا هستند.

برای پیاده‌سازی نرم‌افزارهای اندرویدی، گوگل SDK[1] را ارائه کرده‌است که کتابخانه‌ها و ابزار زیادی را برای برنامه نویسان فراهم آورده است. یکی از این ابزارها ADB[2] shell است. از این ابزار برای برقراری ارتباط با سیستم‌عامل اندروید از طریق ترمینال[3] استفاده می‌شود. محل ذخیره‌سازی این ابزار در <SDK_Path> \sdk\platform-tools\adb.exe است. از طریق دستوراتی که این ترمینال فراهم می‌آورد می‌توان عملیات‌های مورد نیاز از قبیل تغییر تنظیمات دسترسی به فایل‌ها، کپی کردن فایل‌ها از گوشی به رایانه، نصب یک نرم‌افزار روی گوشی و ... را انجام داد. که در جدول شماره یک چند مورد از دستورات آورده شده است.[2]

[1] Software Development Kit

[2] Android Debuge Bridge

[3] Terminal

دسترسی به shell اندروید	adb shell
کپی کردن فایل از گوشی به رایانه	adb push <path in pc> <path in phone>
نصب example.apk از رایانه	adb install example.apk

جدول ۱ دستورات adb shell

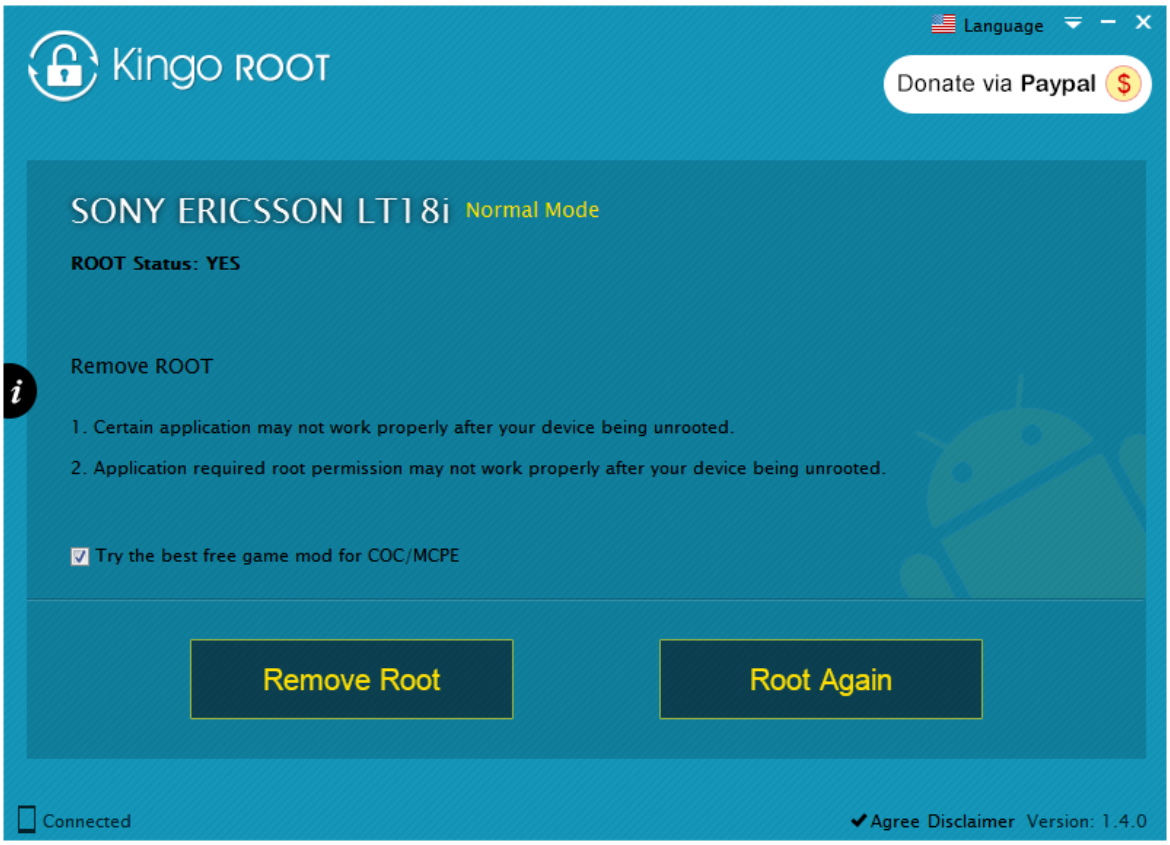
از جمله اطلاعاتی که در سیستم‌عامل اندروید اهمیت دارند، اطلاعات ذخیره‌شده در پایگاه‌داده Sqlite هستند فایل پایگاه‌داده در مسیر /data/data/<app folder>/databases/ ذخیره شده است. برای دسترسی به این فایل‌ها و تغییر تنظیمات مربوط به سطح دسترسی به این فایل‌ها نیاز است که کاربر ممتاز[۱] این تغییرات را انجام دهد. دستیابی به سطح کاربری ممتاز را اصطلاحا روت[۲] کردن سیستم گویند. پس اولین قدم برای دستیابی به فایل‌های پایگاه‌داده، روت کردن سیستم است.

در این پروژه برای روت کردن سیستم از ابزار Kingo Root[۳] استفاده شده‌است. این ابزار قادر است سیستم‌های اندرویدی با پلتفرم‌های مختلف و سیستم‌عامل‌های اندروید با ورژن‌های مختلف را روت کند. این ابزار بروی سیستم‌عامل ویندوز و اندروید ارائه شده‌است. نمایی از این ابزار در شکل‌های ۳و ۴ قابل مشاهده هستند.

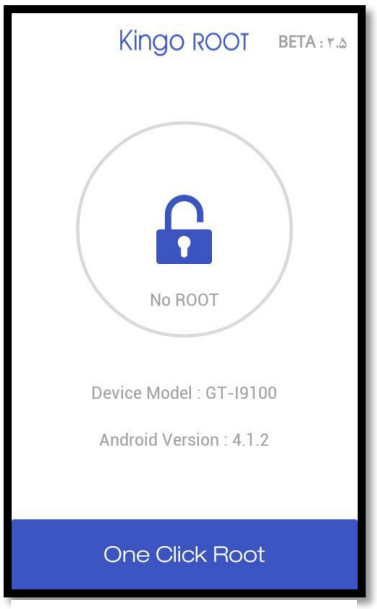
[۱] Privileged user

[۲] Root

[۳] http://www.kingoapp.com/



شکل ۴ نمایی از ابزار Kingo Root تحت ویندوز



شکل ۳ نمایی از ابزار Kingo Root تحت اندروید

بعد از روت کردن سیستم، از طریق دستور chmod 777 <path> دسترسی به فایل‌های پایگاه‌داده تنظیم می‌شود. سپس از طریق دستور push فایل‌ها کپی می‌شوند. برای اجرای این دستورات کتابخانه Android lib استفاده‌شد که امکانات اتصال به گوشی و اجرای دستورات در ترمینال را فراهم می‌کند.

از این کتابخانه در افزونه[1] اندروید استفاده می‌شود که در فصل پنجم به تفصیل چگونگی استفاده و پیاده‌سازی آن بررسی خواهد شد.[1]

[1] Plugin

۳

فصل سوم

بررسی ساختار پایگاه‌داده Sqlite

در این فصل به صورت اجمالی ساختار پایگاه‌داده Sqlite مورد بررسی قرار می‌گیرد. پایگاه‌داده Sqlite شامل یک فایل اصلی و یک فایل ژورنال[۱] است که در ادامه ساختار این دو فایل بررسی می‌شوند.[3] [4]

۱.۳ ساختار فایل اصلی پایگاه‌داده Sqlite

۱.۱.۳ صفحه‌ها[۲]

فایل اصلی پایگاه‌داده شامل تعدادی از صفحه‌هاست. اندازه هر صفحه توانی از دو، میان ۵۱۲ تا ۶۵۵۳۶ بایت می‌تواند باشد. اندازه صفحه‌ها در هر پایگاه داده در ۲ بایت در آفست[۳] ۱۶ از سرآیند[۴] فایل پایگاه‌داده اصلی قرار می‌گیرد. هر یک از این صفحه‌ها در پایگاه‌داده دارای نقشی هستند که با توجه به اینکه در این پروژه فقط از یکی از این نقش‌ها استفاده شده است و آن صفحات موجود در B-tree جداول پایگاه‌داده است، این گونه صفحات در ادامه بررسی می‌شوند.

---

۱ Journal file

۲ Pages

۳ Offset

۴ Header

۲.۱.۳ سرآیند فایل اصلی پایگاه‌داده

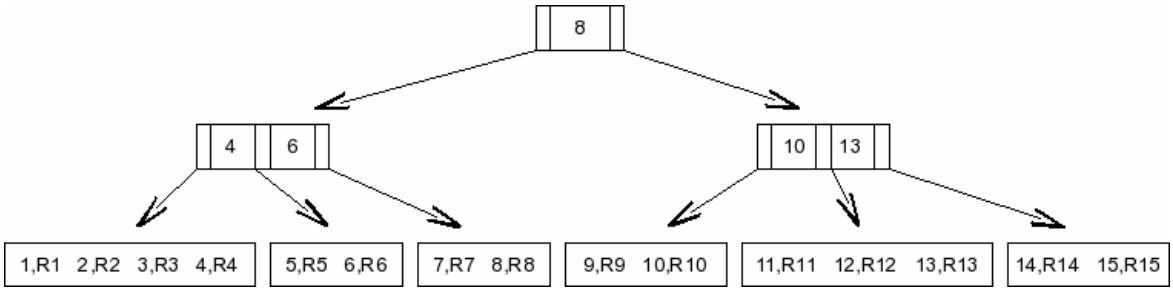
فایل اصلی پایگاه‌داده دارای ۱۰۰ بایت سرآیند است. با توجه به کاربرد این سرآیندها در پروژه حاضر، تعدادی از این سرآیندها در جدول زیر بررسی شده‌اند:

توضیحات	اندازه	آفست
برای تشخیص فایل‌های پایگاه‌داده Sqlite در ابتدای فایل رشته "SQLite format 3\000" وجود دارد.	۱۶	۰
اندازه صفحه‌ها در پایگاه‌داده	۲	۱۶

جدول ۲ سرآیندهای مورد استفاده فایل اصلی پایگاه‌داده [3]

۳.۱.۳ صفحات B-Tree جدولی

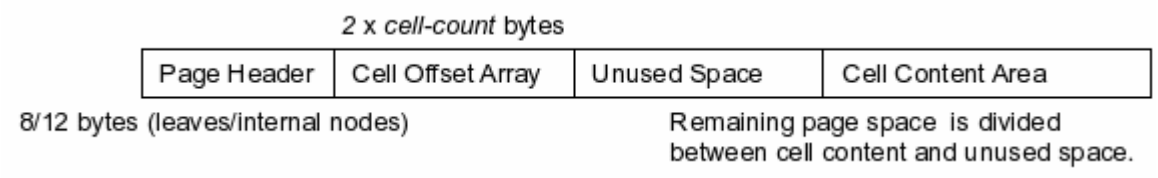
جداول موجود در پایگاه‌داده در ساختار B-Tree ذخیره شده‌اند. شماره صفحه ریشه در جدول sqlite_master ذخیره شده است. ساختار صفحات در B-tree در ادامه ذکر شده‌است. ساختار صفحات میانی و برگ مشابه هم هستند با اندکی تفاوت، که در ادامه بررسی می‌شوند.



شکل ۵ نمایش جداول پایگاه‌داده در ساختار B-Tree[4]

۱.۳.۱.۳ ساختار صفحات B-Tree جدولی

همان‌طور که در شکل ۶ مشاهده می‌شود صفحات در B-tree جدولی از چهار بخش تشکیل شده‌اند: سرآیند صفحه، آرایه آفست‌های سلول‌ها، فضای خالی و فضای قرار گیری سلول‌ها. سرآیندهای صفحات در جدول۳ آمده است.



شکل ۶ ساختار صفحات B-Tree جدولی [4]

توضیحات	اندازه	آفست
پرچم صفحه B-Tree	۱	۰
آفست اولین بلوک در لیست بلوک‌های فضای آزاد. در صورت صفر بودن لیست خالی است.	۲	۱
تعداد سلول‌های موجود در صفحه	۲	۳
آفست بایت اول فضای سلول‌ها	۲	۵
تعداد بایت‌های آزاد پراکنده[۱] در صفحه	۱	۷
شماره صفحه راست‌ترین اشاره‌گر در درخت	۴	۸

جدول ۳ مشخصات سرآیندهای موجود در صفحه‌های B-Tree جدولی [3]

با توجه به توضیحات موجود در جدول بالا ذکر دو نکته لازم است:

- اول اینکه فضاهای آزاد میان فضای در حال استفاده اگر حجم کمتر مساوی ۳ بایت داشته باشند به عنوان بایت‌های آزاد پراکنده تلقی می‌شوند.
- دوم اینکه اگر این فضاهای آزاد بیشتر از ۳ بایت باشند به عنوان بلوک‌های آزاد در نظر گرفته می‌شوند و در یک لیست‌پیوندی قرار می گیرند. آفست اولین بلوک در آفست ۱ از سرآیند صفحه قرار می گیرد. در هر گره دو بایت اول، آفست گره بعد نسبت به آفست شروع صفحه و دو بایت دوم اندازه بلوک را مشخص می‌کنند. در صورت صفر بودن دو بایت اول گره کنونی گره آخر خواهد بود.

[۱] Fragmented free spaces

۲.۳.۱.۳ تفاوت صفحه‌های جدولی داخلی و برگ

- در صفحه‌های داخلی بایت پرچم ۵ و در صفحه‌های برگ ۱۳ است.
- ساختار و محتوای سلول‌ها متفاوت است.

ساختار سلول‌ها در صفحه‌های داخلی در شکل ۷ آمده‌است. قسمت دوم سلول از نوع Var Int[۱] است.

Child page number	Integer Value
4 bytes	1-9 bytes

شکل ۷ ساختار سلول‌های صفحه‌های داخلی B-Tree جدولی [4]

در مورد صفحات برگ ساختار سلول بستگی به اندازه اطلاعات دارد که در صورت کوتاه بودن شکل ۸ و در صورت طولانی بودن شکل ۹ خواهد بود. در صورت طولانی بودن، اطلاعات در قالب لیست‌پیوندی ذخیره می‌شوند. که ساختار هر گره میانی در لیست، در شکل ۱۰ آمده است.

Record Size	Key Value	Database Record
1-9 bytes	1-9 bytes	*Record-size* bytes

شکل ۸ ساختار سلول‌ها با داده‌های کوتاه در صفحات برگB-Tree جدولی [4]

Record Size	Key Value	Database Record Prefix	Overflow page number
1-9 bytes	1-9 bytes	*local-size* bytes, where *local-size* is as defined above	4 bytes

شکل ۹ ساختار سلول‌ها با داده‌های طولانی در صفحات برگB-Tree جدولی [4]

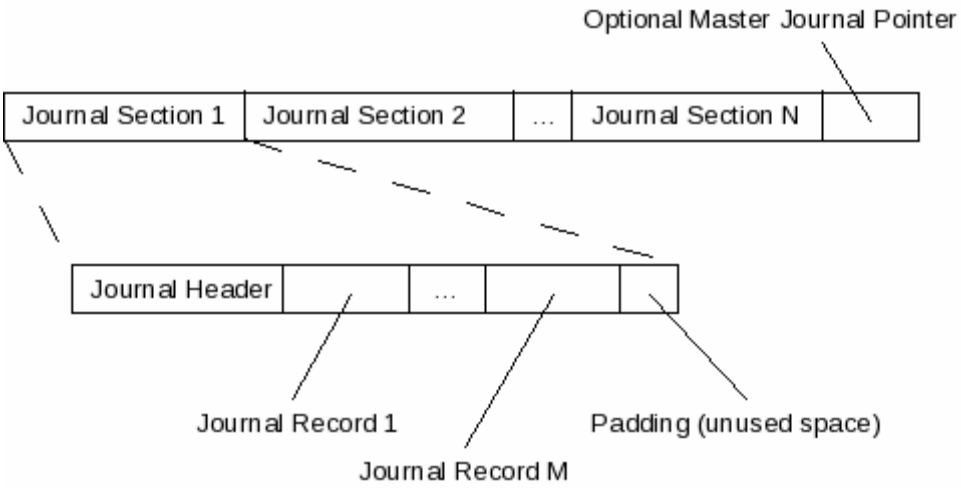
Next Overflow Page number	Record Data
4 bytes	Remaining space

شکل ۱۰ ساختار گره‌های میانی در لیست‌پیوندی مربوط به داده‌های طولانی در صفحات برگ B-Tree جدولی [4]

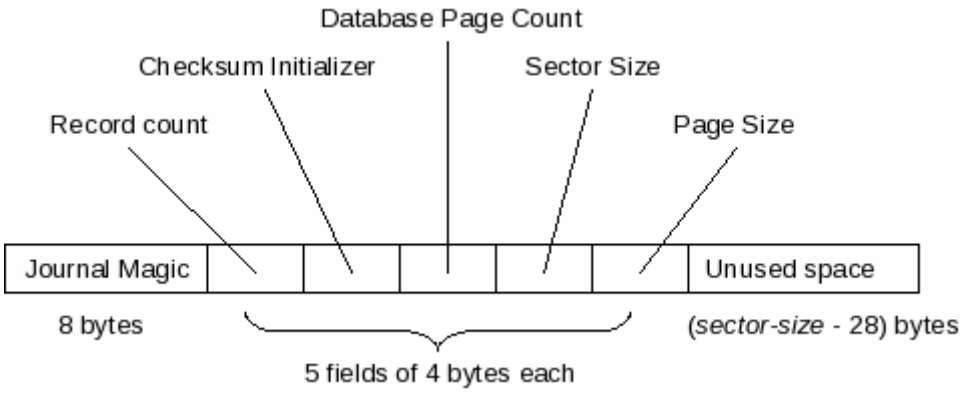
[۱] Variable length integer توضیحات مربوط به این نوع ساختار در مرجع [4] آمده است.

۲.۳ ساختار فایل ژورنال

در پایگاه‌داده Sqlite علاوه بر فایل اصلی پایگاه‌داده فایل دیگری موسوم به فایل ژورنال استفاده می‌شود. نام فایل ژورنال با اضافه شدن رشته "-jornal" به اسم فایل اصلی ایجاد می‌شود. هنگام اجرای هر تراکنش[۱] صفحه‌ای که تغییرات درآن اعمال می‌شود، در فایل ژورنال کپی می‌شود تا در صورت ایجاد هر مشکلی اطلاعات قابل بازگشت باشند. ساختار کلی فایل ژورنال در شکل ۱۱ آمده است. در ادامه ساختار سرآیندها و رکورد[۲]ها در شکل های ۱۲ و ۱۳ آمده است.



شکل ۱۱ ساختار کلی فایل ژورنال [4]

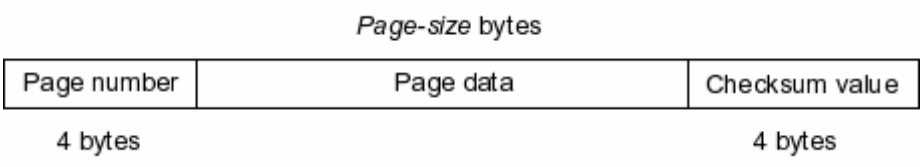


شکل ۱۲ ساختار سرآیندهای فایل ژورنال [4]

---

۱ Transaction

۲ Record



شکل ۱۳ ساختار رکوردهای فایل ژورنال [4]

در صورتی که صفحه‌ای در فایل ژورنال معتبر باشد، یعنی تغییرات در فایل اصلی اعمال نشده باشد، سرآیندهای فایل ژورنال معتبر خواهد بود. در صورت عدم وجود چنین صفحه‌ای پایگاه‌داده تمام بایت‌های موجود در سرآیند را صفر خواهد کرد. از جمله قسمت‌های مهم در سرآیند فایل ژورنال قسمت Journal magic است که برای اعتبار سنجی فایل ژورنال از آن استفاده می‌شود و دیگری اندازه صفحه‌هاست که در روش‌های بازیابی مورد استفاده قرار می‌گیرد. علاوه بر آن اندازه سکتورهاست که مقدار پیش‌فرض آن ۵۱۲ بایت است. ساختار فایل اینگونه است که فایل از تعدادی سکتور تشکیل شده است اگر اندازه سکتور به اندازه رکوردها بخش‌پذیر نباشد، فضای باقی مانده به عنوان فضای بلااستفاده تلقی می‌شود. در خواندن سکتورها باید به این فضاها توجه شود.

اگر بخواهیم ساختار فایل ژورنال را جمع بندی کنیم می توان گفت که بعد از ۲۸ بایت سرآیند رکوردها قرار می‌گیرند که خود شامل شماره صفحه و صفحه تغییریافته و در ادامه ۴ بایت checksum است.

۴

فصل چهارم

روش‌های بازیابی اطلاعات پاک‌شده و تغییریافته از پایگاه‌داده Sqlite

در این فصل، ۳ روش برای بازیابی اطلاعات پاک‌شده و تغییریافته ارائه می‌شود که ۲ روش بر اساس فایل اصلی و یک روش بر اساس فایل ژورنال است.

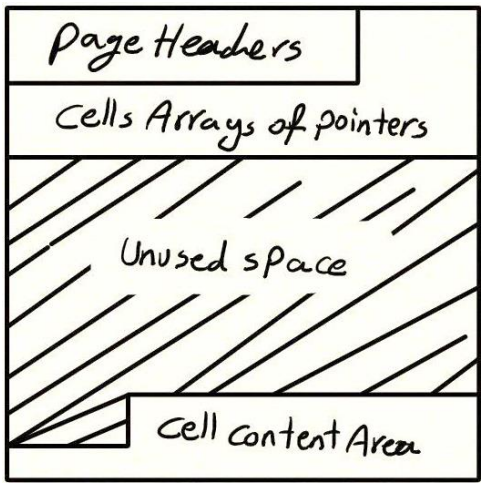
۱.۴ بازیابی بر اساس فایل اصلی پایگاه‌داده

با توجه به ساختار پایگاه‌داده Sqlite که در فصل گذشته بررسی شد، می‌توان به این نتیجه رسید که از ۲ روش می‌توان اطلاعات پاک‌شده یا تغییر پیدا کرده را بازیابی کرد:

۱.۱.۴ بازیابی از طریق فضای بلااستفاده

با توجه به شکل ۱۴ فضای بلااستفاده که هاشور خورده است می‌تواند مستعد حضور داده‌های پیشین باشد. از آنجا که این فضا میان آرایه اشاره‌گرها به سلول‌ها و فضای محتوای سلول‌ها قرار دارد و صفحات با پرچم ۱۳ دارای سلول‌های محتوی داده هستند، می‌توان نتیجه گرفت که در صورت پاک شدن یک رکورد از پایگاه‌داده اشاره‌گر سلول آن از آرایه حذف شده و اطلاعات آن جز فضای بلااستفاده قرار می‌گیرد. پس با استخراج این فضاها می‌توان یک سری از داده‌های پیشین را بازیابی کرد. برای این کار باید B-tree حاوی هر جدول را پیمایش کرد تا با رسیدن به برگ‌ها بتوان فضای بلااستفاده را استخراج کرد. در پروژه این فضاها با نام فضاهای تخصیص‌نیافته[۱] یادشده‌اند.

[۱] Unallocated spaces



شکل ۱۴ ساختار صفحات B-Tree جدولی

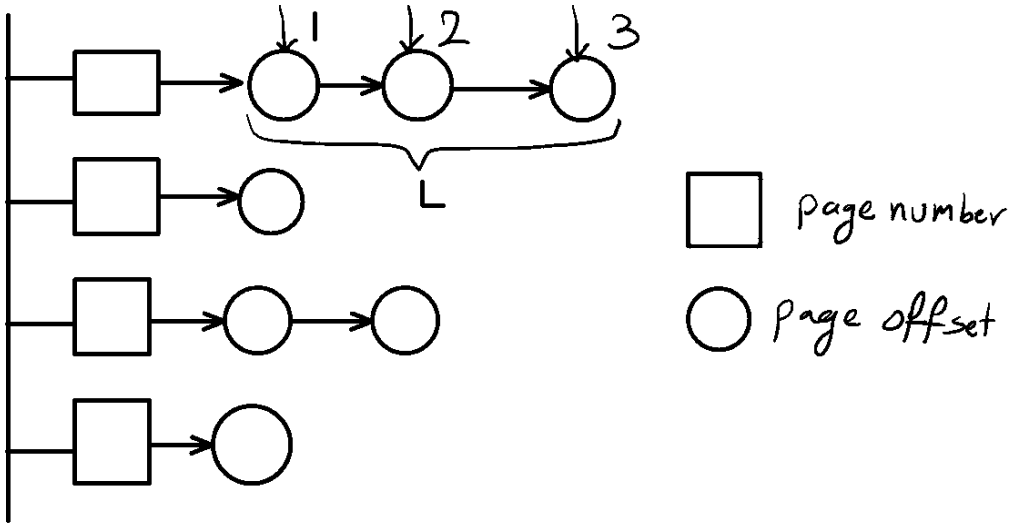
۲.۱.۴ بازیابی از طریق لیست بلوک‌های آزاد

در صورتی که اشاره‌گر به سلولی که میان سلول‌های فعال دیگر قرار دارد، از آرایه اشاره‌گرها حذف شود، فضای آن سلول به عنوان فضای آزاد تلقی می‌شود. در صورتی که این فضا بیشتر از ۳ بایت باشد (که معمولا در مورد سلول‌های شامل رکوردها همین طور است) همان طور که در فصل گذشته بیان شد، جز لیست‌پیوندی بلوک‌های آزاد قرار می‌گیرد. بنابراین محل دوم برای بازیابی اطلاعات این لیست خواهد بود. با پیمایش B-Tree جدولی و دستیابی به صفحه‌های برگ (با پرچم ۱۳) می توان اشاره‌گر به ابتدای این لیست را دریافت و سپس با توجه به ساختار آن که در فصل گذشته ذکر شد آن را پیمایش کرد و داده‌های آن را استخراج نمود.

۲.۴ بازیابی بر اساس فایل ژورنال پایگاه‌داده

همان طور که در فصل گذشته کاربرد فایل ژورنال بررسی شد، می توان دریافت که هر تغییری در پایگاه‌داده اعم از حذف یا به روزرسانی باعث می‌شود یک کپی از صفحه مورد تغییر، قبل از تغییر در این فایل ذخیره شود. این صفحه‌ها از نظر پایگاه‌داده معتبر نیستند ولی برای بازیابی اطلاعات می‌توانند مفید باشند. برای بازیابی این اطلاعات، فایل ژورنال بر اساس ساختار ذکر شده در فصل گذشته خوانده می‌شود و صفحات با پرچم ۱۳ که صفحات برگ در B-Tree هستند همراه با شماره آنها استخراج می‌شوند. در

میان این صفحات ممکن است که صفحات با شماره یکسان چند بار تکرار شده باشند که آفستی که این صفحات در آنها وجود دارد در یک لیست قرار خواهد گرفت. همان طور که در شکل ۱۵ آمده است



شکل ۱۵ ساختمان داده استفاده شده برای نگهداری آفست و شماره صفحات موجود در فایل ژورنال

آفست صفحه‌ها همراه با شماره آنها در ساختار موجود در شکل ذخیره شده است. حال برای استخراج داده‌ها از این روش استفاده می‌شود که به ازای L که طول بلندترین لیست از آفست‌هاست از فایل پایگاه‌داده کپی ایجاد می‌شود. سپس صفحه‌های موجود در هر ستون در صورت وجود، به طور مثال ستون 1، با توجه به رابطه page offset = (page number - 1)* page size در فایل پایگاه‌داده کپی شده، جایگزین می‌شوند. سپس این فایل به وسیله ابزار sqldiff.exe که توسط Sqlite برای مقایسه پایگاه‌داده‌ها ارائه شده است، با فایل اصلی پایگاه‌داده مقایسه می‌شود. خروجی این ابزار به این شرح است که به تفکیک جدول، پرس وجوهایی[۱] که مشخص می‌کند چه رکوردی از پایگاه‌داده تولیدی نسبت به پایگاه‌داده اصلی پاک[۲]، به‌روز[۳] یا اضافه[۴] شده است را در خروجی نشان می‌دهد. پرس‌وجوهای اضافه‌کننده چون در خروجی پایگاه‌داده فعلی قابل مشاهده هستند، مفید نخواهند بود ولی پرس‌وجوهای

---

۱ Query

۲ Delete

۳ Update

۴ Insert

پاک‌کننده و به‌روزکننده با تغییر به پرس‌وجوی انتخاب[1] از پایگاه داده تولیدی، می تواند رکوردهایی که در پایگاه‌داده فعلی موجود نیست را به ما نشان دهد. شکل زیر خروجی این ابزار را در حالت خلاصه نشان می‌دهد.

این روال به ازای تمام ستون‌ها در ساختمان‌داده شکل ۱۵ تکرار شده و رکوردهای حاصله تجمیع شده و در خروجی نشان داده می‌شود.

```
addr: 0 changes, 0 inserts, 0 deletes, 0 unchanged
android_metadata: 0 changes, 0 inserts, 0 deletes, 1 unchanged
attachments: 0 changes, 0 inserts, 0 deletes, 0 unchanged
canonical_addresses: 40 changes, 45 inserts, 207 deletes, 0 unchanged
drm: 0 changes, 0 inserts, 0 deletes, 0 unchanged
part: 0 changes, 4 inserts, 0 deletes, 0 unchanged
pdu: incompatible schema
pdu_recipient_threads: missing from first database
pending_msgs: 0 changes, 0 inserts, 0 deletes, 0 unchanged
rate: 0 changes, 0 inserts, 0 deletes, 0 unchanged
raw: 0 changes, 32 inserts, 0 deletes, 0 unchanged
semc_metadata: missing from first database
semc_threads: missing from first database
sms: incompatible schema
sqlite_sequence: 2 changes, 1 inserts, 0 deletes, 0 unchanged
sr_pending: 0 changes, 0 inserts, 0 deletes, 0 unchanged
threads: 35 changes, 52 inserts, 227 deletes, 0 unchanged
words_content: 1576 changes, 81 inserts, 2361 deletes, 0 unchanged
words_segdir: 0 changes, 18 inserts, 28 deletes, 0 unchanged
words_segments: 0 changes, 273 inserts, 114 deletes, 0 unchanged
```

شکل ۱۶ خروجی مقایسه دو پایگاه‌داده که پایگاه‌داده اول پایگاه‌داده حاصل از فایل ژورنال و پایگاه‌داده دوم پایگاه داده اصلی است، به صورت خلاصه آمده است.

[1] Select

۵

فصل پنجم

پیاده‌سازی نرم‌افزار

برای پیاده‌سازی نرم‌افزار از زبان برنامه نویسی C# با رویکرد شی‌گرا[۱] استفاده‌شد. همچنین مدل فرایند[۲] استفاده شده در این نرم‌افزار مدل فرایند آبشاری[۳] است که در ادامه به بررسی آن و همچنین نمودارهای لازم برای تحلیل نرم‌افزار پرداخته خواهد شد. علاوه بر آن نرم‌افزار کنونی دارای قابلیت توسعه‌پذیری است که روند اعمال و پیاده‌سازی آن در نرم‌افزار در ادامه توضیح داده خواهد شد. در نهایت واسط گرافیکی پیاده‌سازی شده برای نرم‌افزار مورد بررسی قرار می‌گیرد.

۱.۵ تحلیل و طراحی نرم‌افزار

مدل آبشاری یک مدل ترتیبی توسعه و تولید نرم افزار است و در آن مراحل تولید به شکل یک جریان مداوم متمایل به سمت پأیین می‌باشد. همانند یک آبشار که شامل فازهای تحلیل خواسته‌ها، طراحی، پیاده‌سازی[۴]، آزمودن[۵]، یکپارچه سازی[۶] و دادن محصول به بازار می‌شود. مدیریت و مراحل تکمیل پروژه در این مدل فرایند به سادگی قابل پیاده‌سازی است. زیرا در مرحله اول که مرحله بررسی نیازمندی‌های پروژه می‌باشد، مشتری و تیم برنامه نویسی طی چند جلسه به بررسی نیازمندی‌ها و

---

۱ Object oreinted

۲ Process model

۳ Waterfall

۴ Implementation

۵ Test

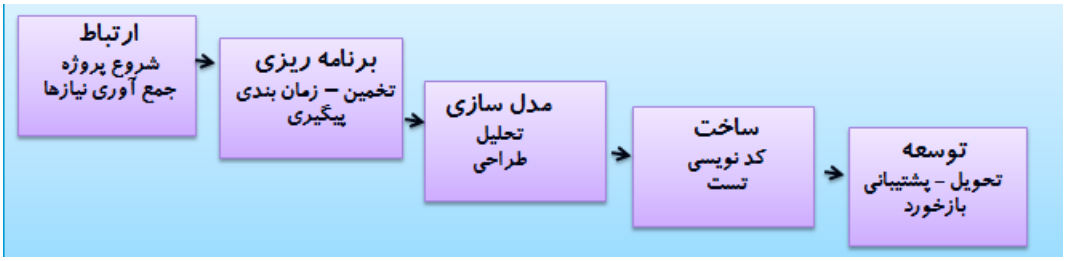
۶ Integration

خواسته‌های پروژه می‌پردازند. پس از آن نوبت به مرحله طراحی می‌رسد، در مرحله‌ی طراحی افراد طرح کلی پروژه را می‌ریزند و جزییات پیاده‌سازی مشخص می‌شود. پس از مرحله‌ی طراحی تیم برنامه نویسی خود را برای پیاده‌سازی آماده می‌کند. در این مرحله همه قسمت‌های کد، پیاده‌سازی می‌شوند و در انتهای این مرحله، مرحله یکپارچه سازی را خواهیم داشت که یکی از مشکل‌ترین قسمت‌های انجام پروژه‌ها در این مرحله می‌باشد. زیرا تنوع و گستردگی کار کاملا در این مرحله نقش دارد، هر چه میزان گستردگی کار بالاتر باشد سختی یکپارچه سازی نیز بیشتر خواهد بود.

در این پروژه با توجه به مشخص و ثابت بودن نیازهای نرم‌افزار در ابتدای تعریف پروژه، می‌توان از مدل فرآیند آبشاری استفاده کرد. علاوه بر آن این مدل فرآیند دارای مزیت‌هایی است که استفاده از آن را تایید می‌کند که در ادامه به مزیت‌های این مدل فرایند پرداخته می‌شود.

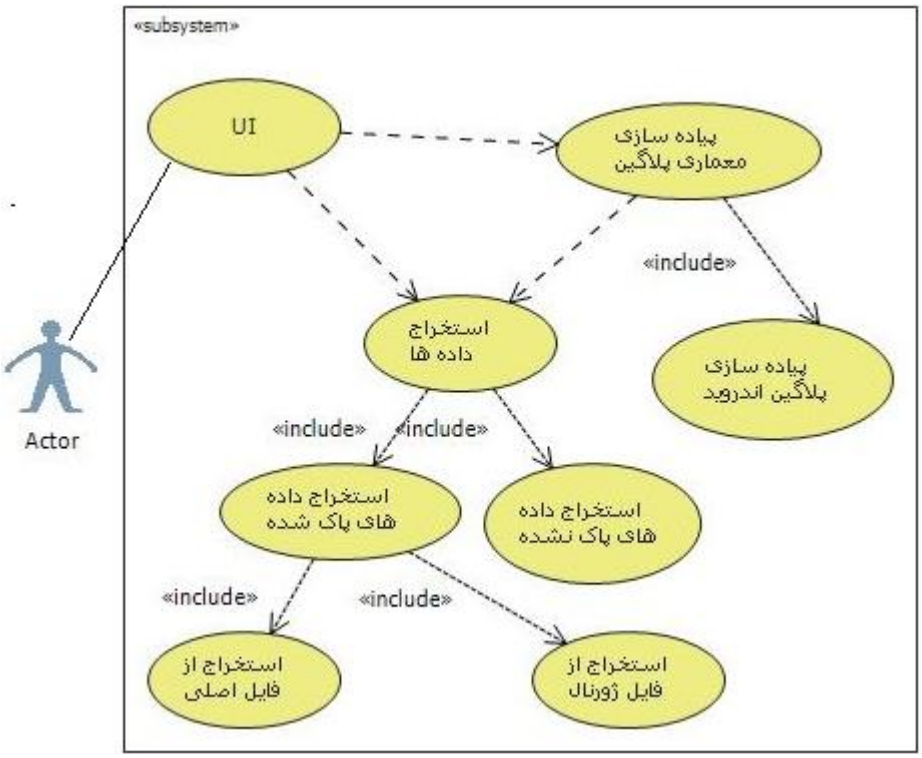
مزیت‌های مدل آبشاری به شرح زیر می‌باشد:

- فهم این مدل ساده‌تر است.
- از نظر تولید مستندات شرایط بهتر و آسان‌تری دارد.
- مراحل قابل کنترل و بررسی می‌باشند.

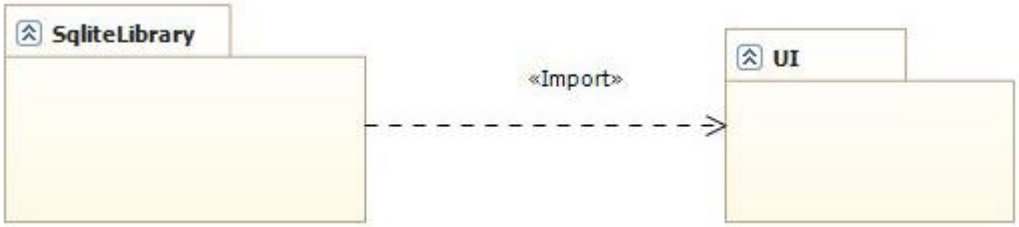


شکل ۱۷ فرآیند تولید نرم‌افزار به صورت آبشاری

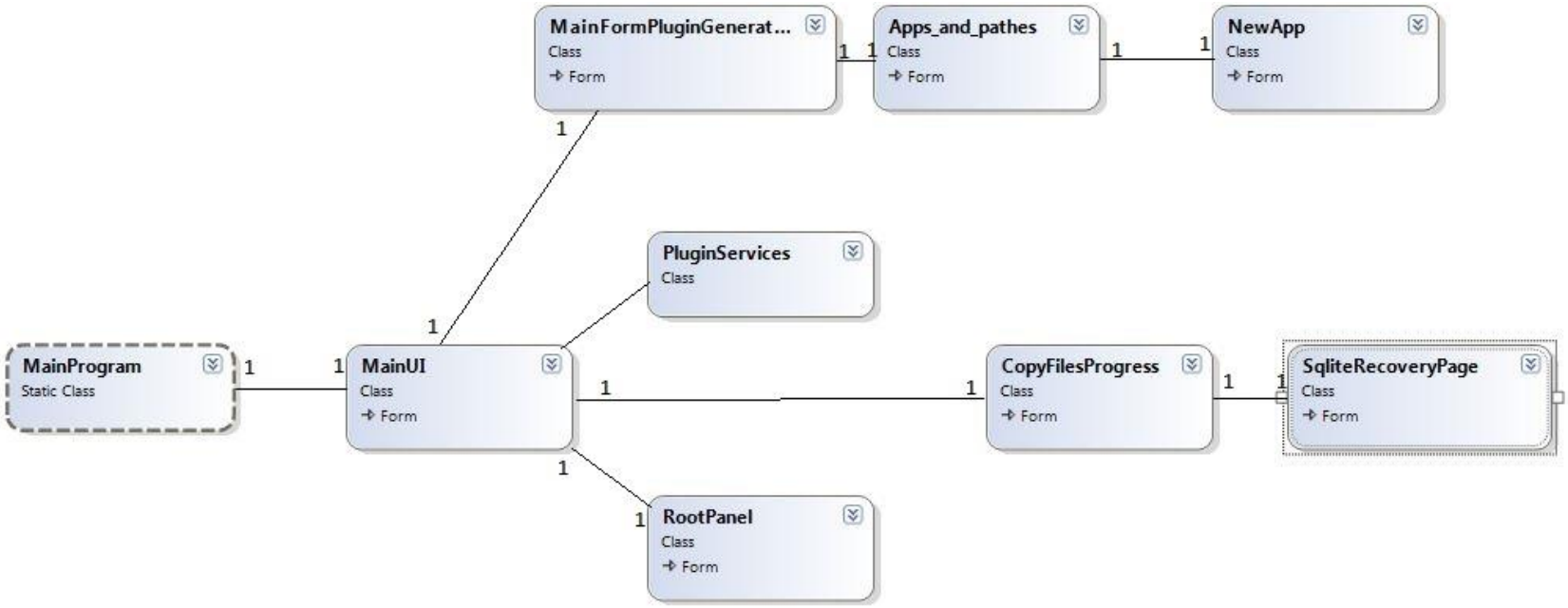
در مرحله تحلیل نرم‌افزار ابتدا نمودار usecase ایجاد می‌شود که در شکل ۱۸ قابل مشاهده است. در ادامه نمودارهای package و class در شکل‌های ۱۹، ۲۰ و ۲۱ آمده‌اند. گراف وابستگی کلاسها هم در شکل ۲۲ آمده است.



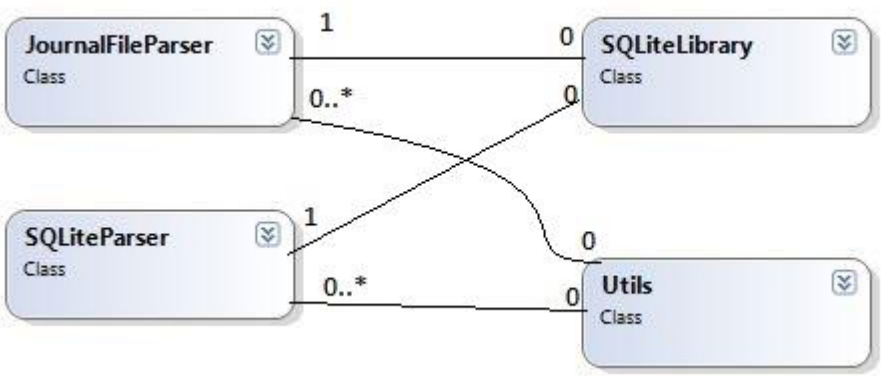
شکل ۱۸ نمودار usecase



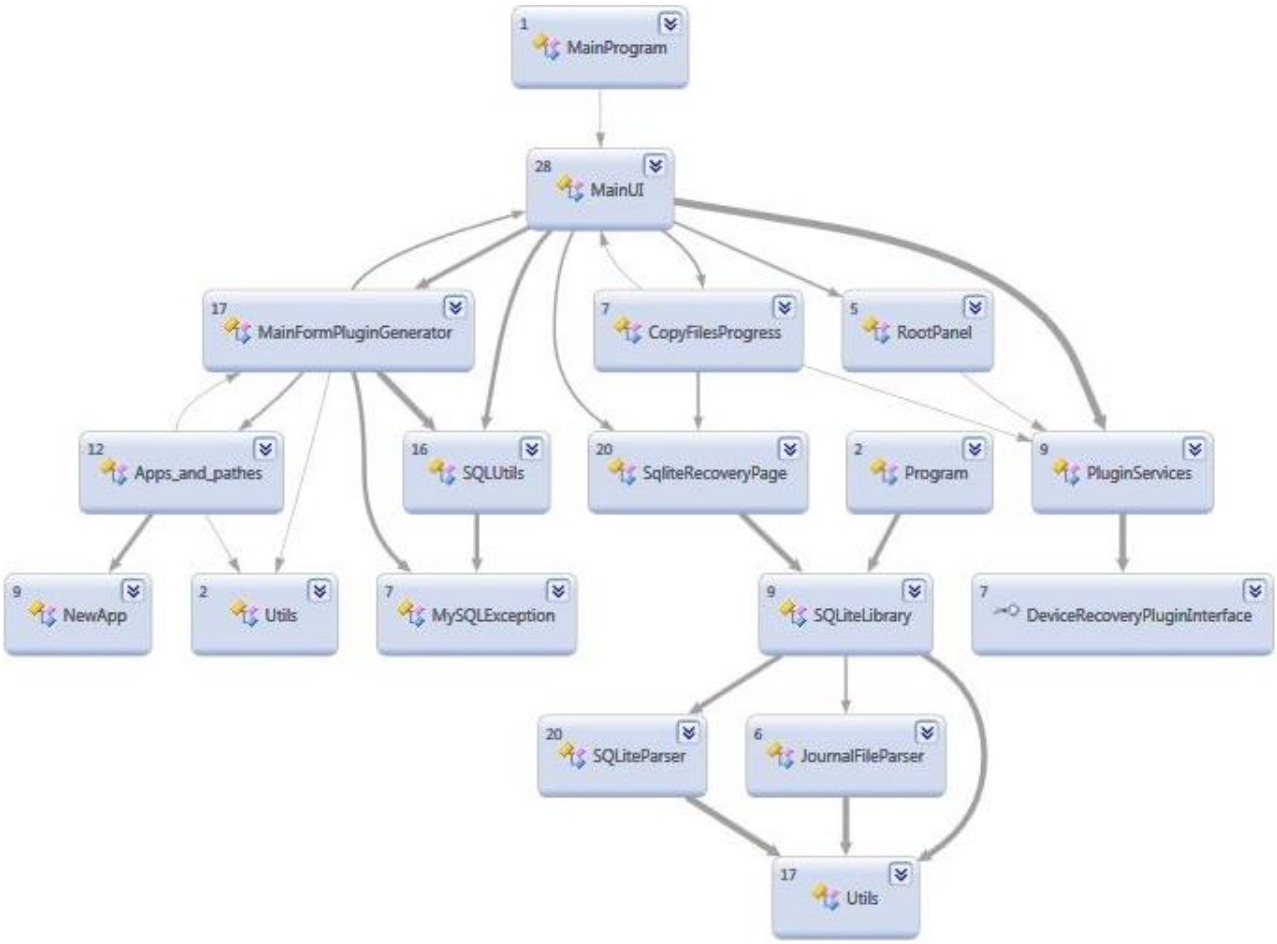
شکل ۱۹ نمودار package



شکل ۲۰ نمودار class کتابخانه UI



شکل ۲۱ نمودار class برای کتابخانه sqlitelibrary



شکل ۲۲ نمودار وابستگی کلاس‌ها

۲.۵ قابلیت توسعه‌پذیری

همان طور که پیش از این ذکر شد پایگاه‌داده Sqlite در پلتفرم‌های مختلف از جمله اندروید، iOS و ویندوز کاربرد دارد. برای اینکه نرم‌افزار حاضر بتواند از همه این پلتفرم‌ها، فایل‌های پایگاه‌داده را بگیرد و پردازش کند، قابلیت توسعه‌پذیری به سیستم اضافه شد. برای این منظور از طریق واسط کاربری که در بخش بعد به تفصیل توضیح داده خواهد شد نام و آدرس ذخیره‌سازی پایگاه‌داده‌ها، آدرس فایل dll (شامل دستورات لازم برای اتصال، کپی کردن و ... به زبان C# است این دستورات به واسطه یک interface باید پیاده‌سازی شوند که کد آن در شکل ۲۳ آمده است. سرانجام این کدها کامپایل شده و فایل dll تولید می‌شود.) و نوع سیستم‌عامل، دریافت شده و یک افزونه تولید می‌شود. این افزونه از طریق سیستم reflection در C# و بارگذاری فایل‌های اسمبلی، فایل dll را در نرم‌افزار بارگذاری کرده و از تابع‌های موجود در interface استفاده می‌کند. برای کنترل اینکه فایل بارگذاری‌شده معتبراست، نوع فایل بارگذاری شده باید از نوع interface باشد یعنی interface باید در آن پیاده‌سازی شده باشد. در این پروژه یک افزونه به منظور اتصال به دستگاه‌های اندرویدی پیاده‌سازی شده است. برای پیاده‌سازی این

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace DevicePluginInterface
{
    public interface DeviceRecoveryPluginInterface
    {
        void copyAppDataBaseFromDevice(string key,string path, string distination);

        bool isDeviceRoot();

        bool rootDevice();

        bool unRootDevice();

        bool isDeviceConnected();

        void refreshDeviceList();

        bool installApp(string path);
    }
}
```

شکل ۲۳ نمایی از کد interface و تابع‌هایی که برای هر افزونه باید پیاده‌سازی شوند.

افزونه از کتابخانه Android Lib استفاده شد که دستورات ADB Shell در آن پیاده‌سازی شده‌اند.

۳.۵ واسط کاربری گرافیکی[۱]

اهمیت ظاهر برنامه و صفحاتی که کاربر توسط آن ها با سیستم در تعامل است، بر کسی پوشیده نیست. در پروژه حاضر به دلیل وجود قابلیت توسعه‌پذیری این مورد اهمیت بیشتری پیدا می‌کند چون واسط کاربری باید به گونه‌ای باشد که کاربر به راحتی بتواند افزونه‌های مورد نیاز خود را تولید کرده و به سیستم اضافه کند. علاوه بر آن واسط کاربری باید امکانات لازم برای دسترسی به اجزای مختلف پایگاه‌داده‌ها را فراهم آورد که این موارد از طریق اضافه کردن Tab، ComboBox و DataGridView فراهم آمده است.

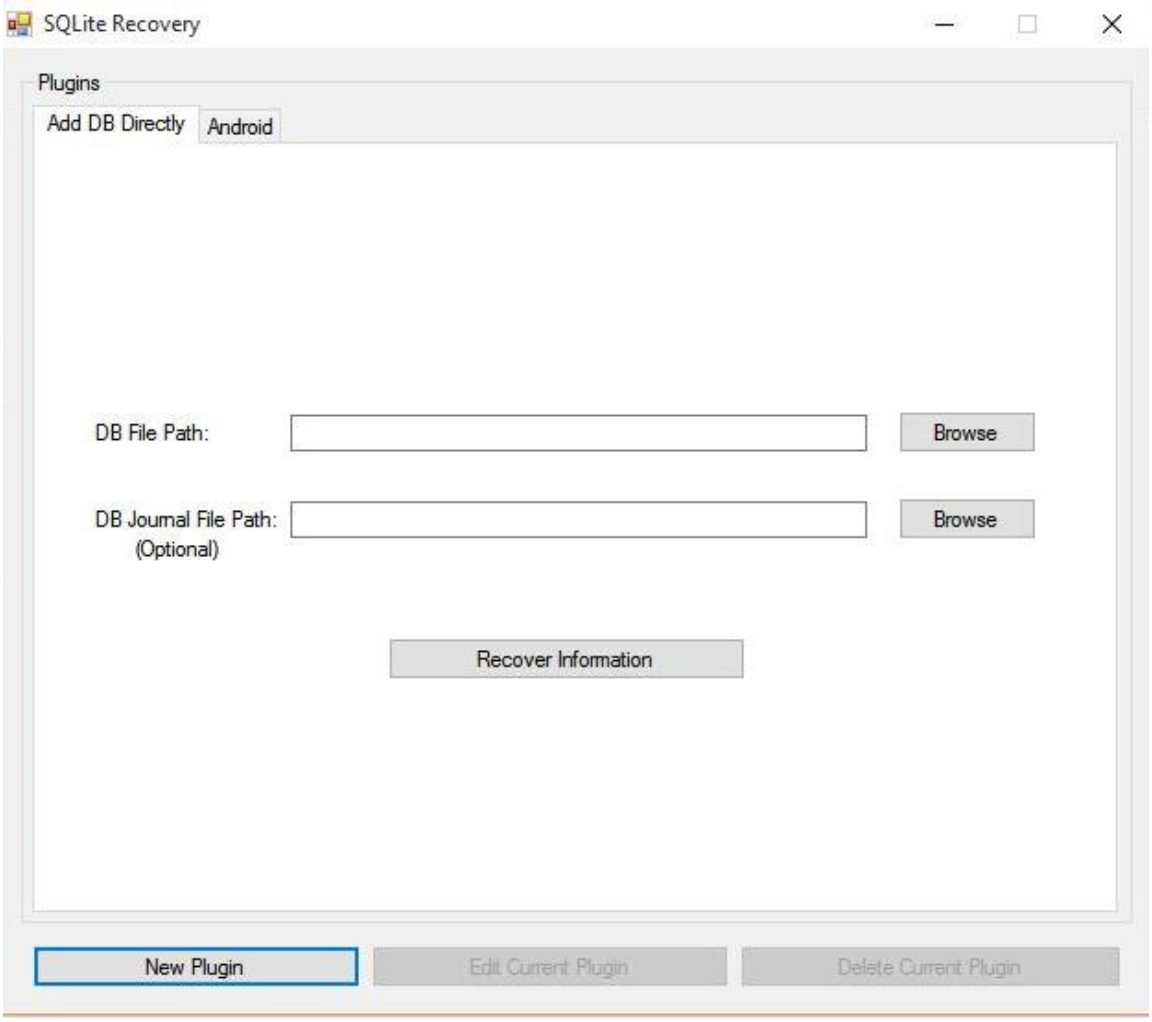
یک واسط کاربری خوب دارای ویژگی‌هایی است که در زیر به برخی از آنها اشاره شده‌است:

- گزینه‌ها و دکمه های موجود در صفحه باید همگون و با سبک یکسان باشند.
- در هنگام تغییر وضعیت برنامه، باید ظاهر نیز متناسب با آن تغییر یابد. یعنی برنامه متناسب با هر فعالیت، بازخورد مناسبی داشته باشد.
- هر گزینه باید کاملا واضح و دارای معنای خاص باشد.
- برای همگی فعالیت‌ها، حالت های پیش فرض در نظر گرفته شود.
- کاربر نیازی به آموزش برای یادگیری کار با رابط کاربری نداشته باشد یا حداقل باشد.
- اجزائی که با یکدیگر مرتبط هستند، در یک گروه‌بندی خاص باشند
- برای حذف یا پاک کردن اطلاعات مهم، تأیید مجدد کاربر دریافت شود.
- امکان تغییر ابعاد صفحه برای کاربر وجود داشته باشد و ضمناً با تغییر ابعاد پنجره برنامه، چینش اجزا در صفحه منظم باقی بماند

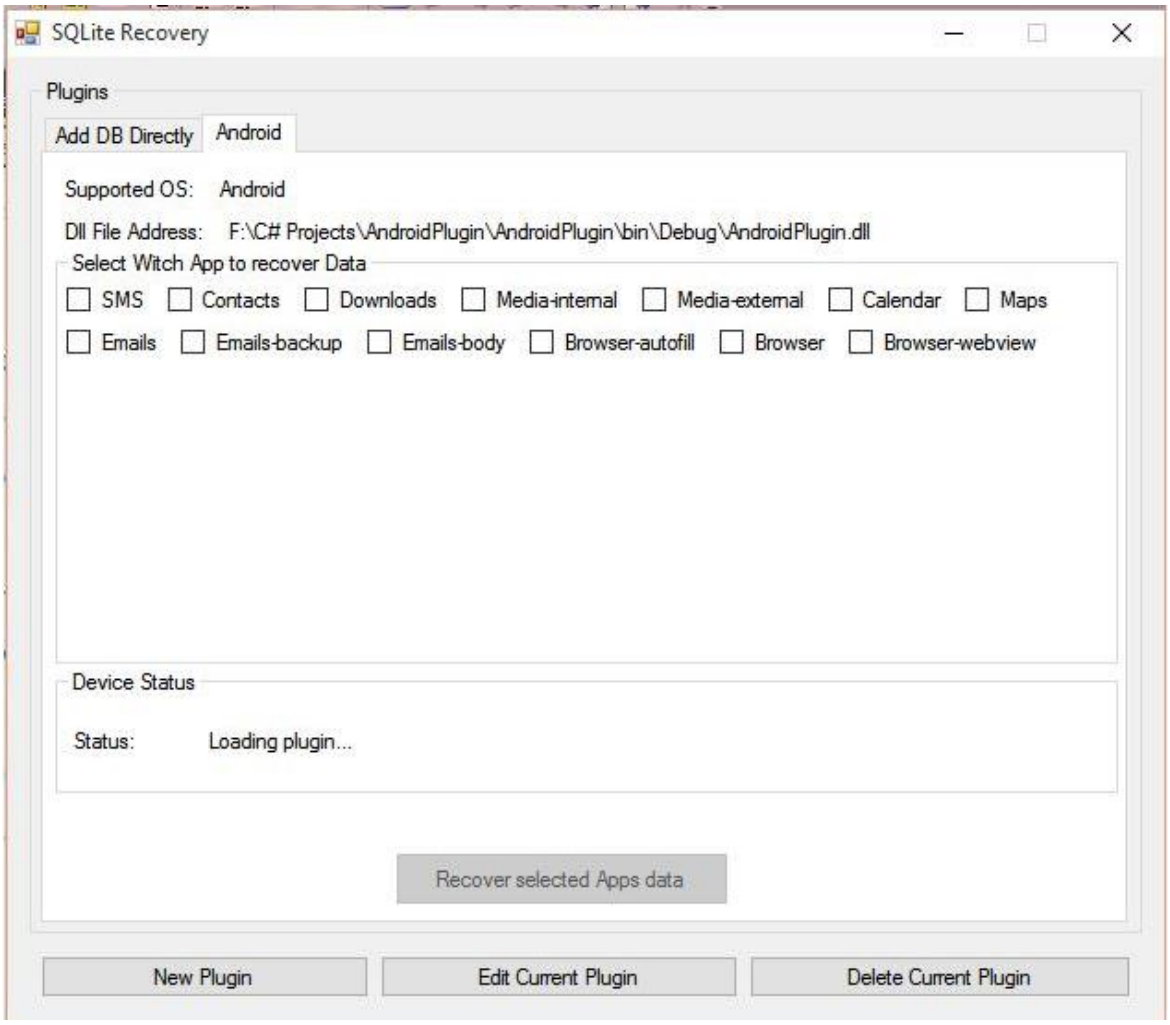
در این پروژه سعی شده‌است موارد بالا رعایت شود. در ادامه نمایی از واسط‌های کاربری پیاده‌سازی شده آمده است.

---

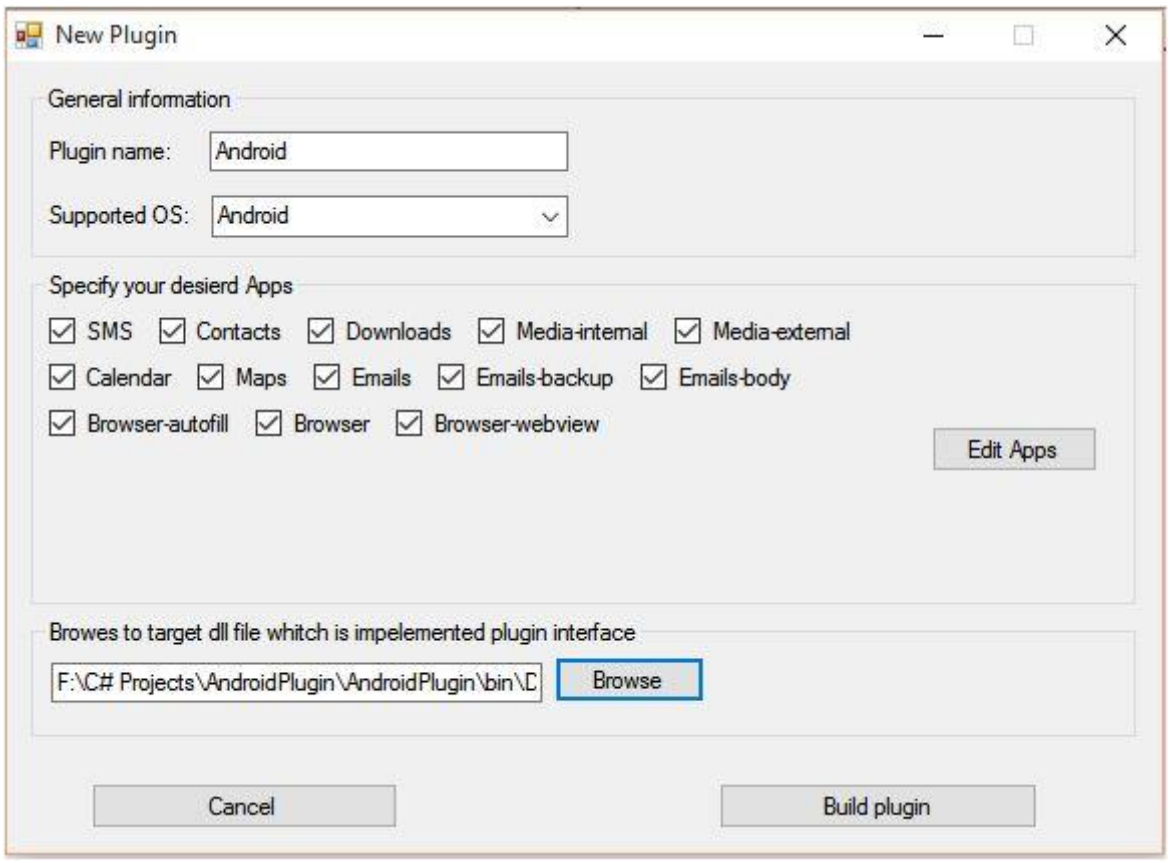
[۱] Graphical user interface(GUI)



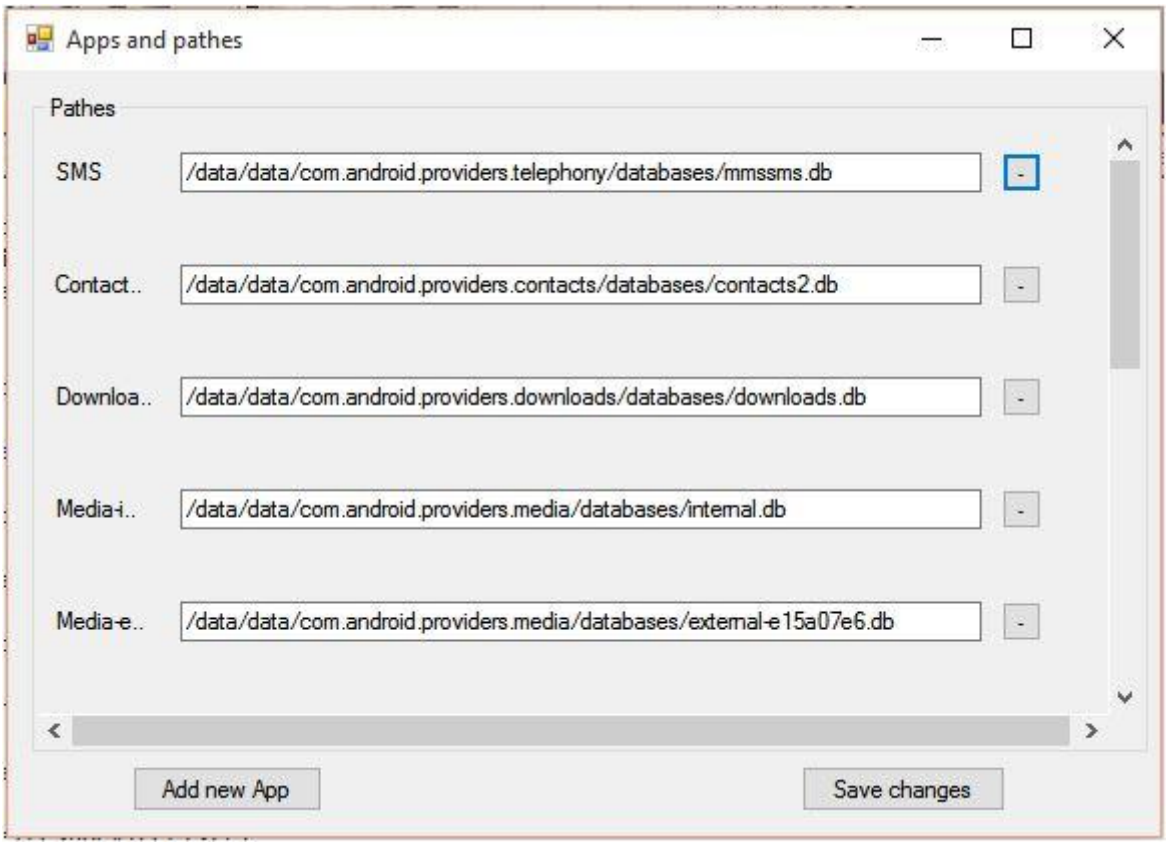
شکل ۲۴ صفحه اصلی برنامه، پردازش پایگاه‌داده‌های موجود در رایانه



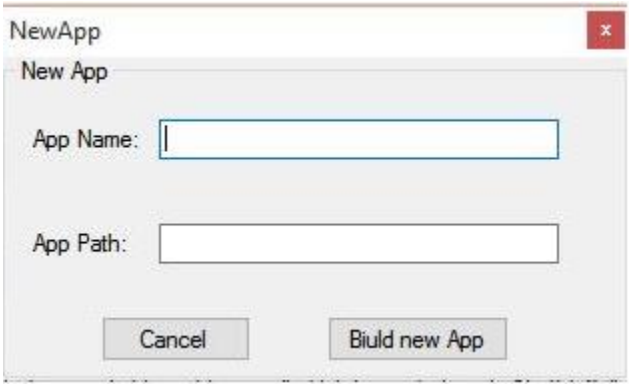
شکل ۲۵ نمایی از یک افزونه



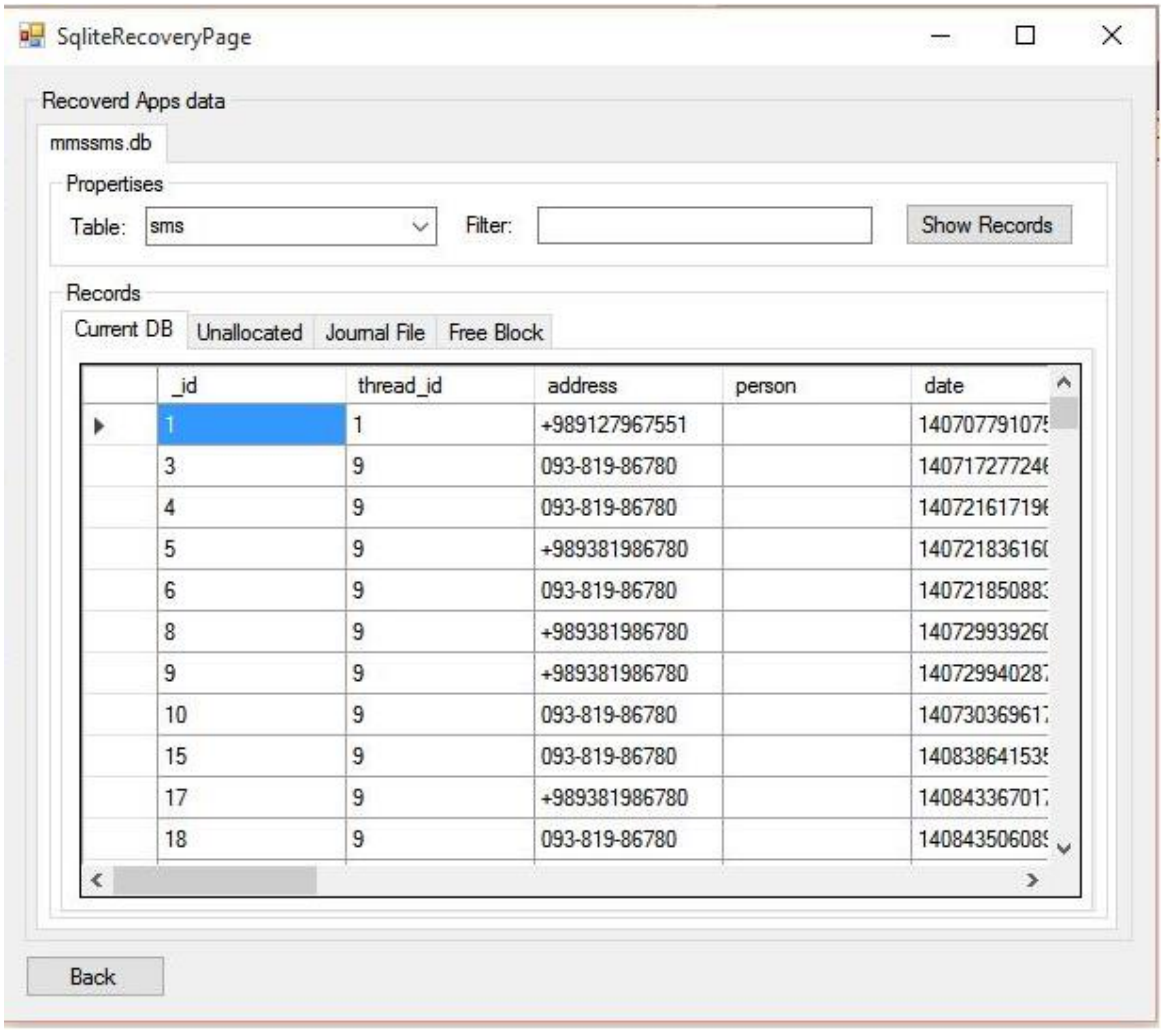
شکل ۲۶ ایجاد افزونه جدید



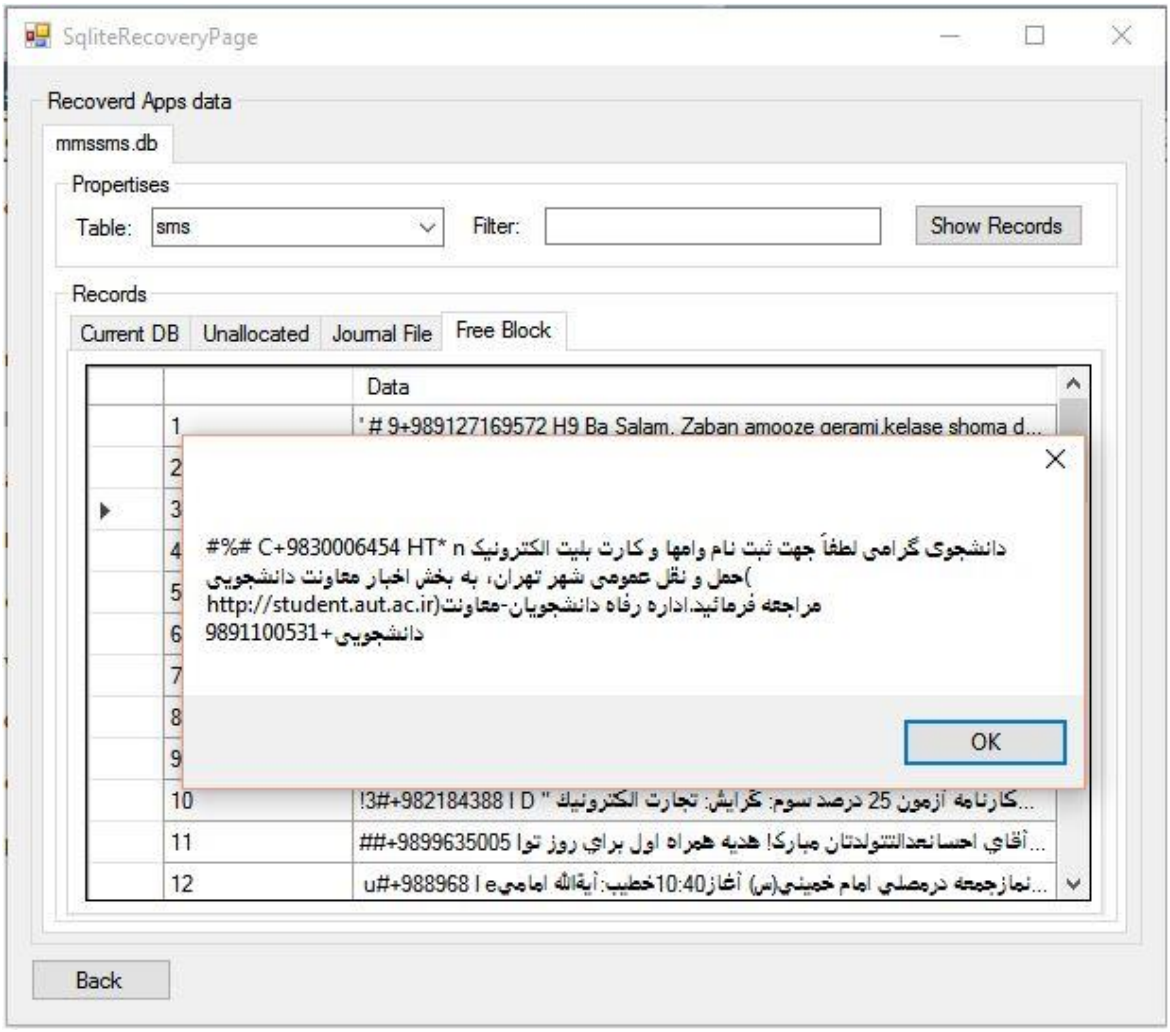
شکل ۲۷ حذف یا اضافه کردن نام و آدرس نرم‌افزار به افزونه



شکل ۲۸ اضافه کردن نام و آدرس نرم‌افزار



شکل ۲۹ صفحه داده‌های پایگاه‌داده‌ها



شکل ۳۰ نمایی از داده‌های پاک‌شده

۶

فصل ششم

جمع‌بندی و کارهای آینده

در فصول گذشته در مورد اهمیت ابزارهای هوشمند و گستردگی این ابزارها میان مردم صحبت شد. اطلاعات ذخیره شده در این ابزارها و ارزش و اهمیت این داده‌ها در کاربردهای استخراج اطلاعات به منظور اثبات جرم توسط پلیس بررسی شد. ساختار و داده‌های ذخیره شده در یکی از پرکاربردترین سیستم‌عامل‌های استفاده شده در ابزارهای هوشمند یعنی اندروید مورد بررسی قرار گرفت که نتیجه آن، کاربرد پایگاه‌داده Sqlite در ذخیره‌سازی داده‌های هر نرم‌افزار بود. البته این پایگاه‌داده در پلتفرم‌های دیگر از جمله iOS و ویندوز نیز کاربرد دارد. روش دستیابی به پایگاه‌داده Sqlite در گوشی‌های هوشمند اندرویدی مورد بررسی قرار گرفت. پس از آن ساختار پایگاه‌داده Sqlite بررسی شد تا به کمک آن روش‌های استخراج و بازیابی داده‌های ذخیره شده در آن توضیح داده شود و در پایان روند تحلیل و پیاده‌سازی نرم‌افزار و پیاده‌سازی ویژگی توسعه‌پذیری و در نهایت واسط کاربری گرافیکی مورد بررسی قرار گرفت.

آنچه که در این پروژه مورد بررسی قرار گرفت بازیابی بخش کوچکی از داده‌های ذخیره شده در ابزارهای هوشمند اندرویدی بود. علاوه بر آن ابزارهای هوشمند اندرویدی دارای طیف گسترده‌ای از نظر نسخه سیستم‌عامل استفاده شده و پلتفرم سخت‌افزاری هستند، که نحوه دسترسی به فایل‌های پایگاه‌داده در آنها ممکن است اندکی متفاوت باشد، که این مورد نیز باید مورد بررسی قرار گیرد. علاوه بر سیستم‌عامل اندروید، سیستم‌عامل iOS و ویندوز نیز باید مورد بررسی قرار گیرند. علاوه بر داده‌های ذخیره شده در Sqlite داده‌های دیگر از جمله عکس‌های ذخیره شده و ... نیز می‌توانند موضوع تحقیق قرار گیرند. البته ذکر این نکته لازم است که پیاده‌سازی ویژگی توسعه‌پذیری این امکان را می‌دهد که پیاده‌سازی موارد بالا و اضافه کردن آنها به نرم‌افزار کنونی راحت‌تر صورت پذیرد.

منابع و مراجع

[1] A. Hoog and J. McCash, Android Forensics. Waltham, MA: Syngress, 2011.

[2] Developer.android.com, 'Download Android Studio and SDK Tools | Android Developers', 2015. [Online]. Available: https://developer.android.com/sdk/index.html. [Accessed: 02- Oct- 2015].

[3] Sqlite.org, 'File Format For SQLite Databases', 2015. [Online]. Available: http://sqlite.org/fileformat2.html. [Accessed: 02- Oct- 2015].

[4] Sqlite.org, 'File Format For SQLite Databases', 2015. [Online]. Available: http://sqlite.org/fileformat2.html. [Accessed: 02- Oct- 2015].

Amirkabir University of Technology
(Tehran Polytechnic)

Computer and Information Technology Engineering Department

B.Sc. Thesis

Title

Design and Implementation of Extensible Software in order to Retrieve Deleted Information from Smart Phones

By
Ehsan Edalat

Supervisor
Dr. Babak Sadeghian

September 2015