



## نرم افزار مدیریت پروژه

### اعضای گروه:

فریبا بوستانی  
رایحه حسینی پور  
نسترن حکیمی  
مهلا راحتی  
سارا متین فر



## فهرست

۲	مقدمه
۲	خلاصه
۲	دامنه
۳	ابعاد محصول
۳	مسائل حقوقی
۴	ویژگی های اصلی
۴	موارد فنی
۴	زمان بندی
۶	ابزار های مورد استفاده
۸	محدودیت ها
۸	پیش فرض ها
۸	متدولوژی
۱۰	معماری مورد استفاده
۱۲	نحوه تست و انواع تست نرم افزار
۱۵	نحوه نصب نرم افزار
۱۶	افراد و نقش ها
۱۶	پشتیبانی



## مقدمه

### خلاصه :

یک پروژه مجموعه‌ای از فعالیت‌ها است که برای دستیابی به منظور یا هدف خاصی انجام می‌گیرد. پروژه‌ها شامل فعالیت‌هایی هستند که باید در تاریخ‌های معین، با هزینه‌هایی معین و کیفیت تعیین شده‌ای به انجام برسند؛ لازمه موفقیت هر پروژه، دستیابی توأم به هر سه عامل زمان، هزینه و کیفیت معین است و خارج شدن هر یک از سه عامل مذکور از حدود تعیین شده، می‌تواند به انجام پروژه‌ای ناموفق و غیراقتصادی منجر شود.

مدیریت پروژه برنامه‌ریزی و هدایت پروژه در چهارچوب زمان، هزینه و کیفیت مشخص به سوی ایجاد نتایج مشخص آن است. مدیریت پروژه فعالیت‌های برنامه‌ریزی، سازماندهی، نظارت بر اجرا و هدایت اجرا را در بر می‌گیرد و سعی دارد تا با استفاده درست از منابع، نتایج مشخص و مورد انتظار را با هزینه توافق شده قبلی در موعد درست خود تحویل دهد.

با توجه به گسترش پروژه‌ها در موضوع‌های مختلف نیاز به نرم‌افزاری که بتوان به کمک آن روند انجام یک پروژه را مدیریت کرد احساس می‌شود. وجود نرم‌افزار مدیریت پروژه بر روی تلفن همراه این امکان را به ما میدهد تا در هر جا و هر زمان با دسترسی به اینترنت از وضعیت پروژه، اخبار مربوط به آن، زمانبندی اجرای کارهای مختلف و ... اطلاع پیدا کنیم.

یکی از مهم‌ترین اهداف این نرم‌افزار زمان‌بندی یک سری از رویدادها و فعالیت است. این نرم‌افزار با گرفتن ورودی می‌تواند اطلاعاتی را درمورد افراد و ذینفعان مختلف فراهم سازند که با استفاده از این اطلاعات می‌توان میزان تلاش مورد نیاز برای به پایان رساندن پروژه را تعیین و تصدیق کرد.

### دامنه :

این نرم‌افزار برای همه افرادی که به نحوی با کارهای گروهی، پروژه‌های سازمانی، پروژه‌های نرم-افزاری و ... سروکار دارند قابل استفاده می‌باشد. امکان مدیریت برنامه‌های شخصی هر فرد، کارهای گروهی دانش‌آموزی، پروژه‌های دانشجویی، پروژه‌های بزرگ سازمانی و به طور کلی کلیه فعالیت



هایی که با عنوان "کار گروهی" و یا "پروژه" مطرح هستند و نیاز به زمانبندی، تقسیم وظایف، اطلاع از میزان پیشرفت پروژه و ... دارند در این نرم افزار وجود دارد.

### ابعاد محصول :

این اپلیکیشن بصورت مستقل از سایر نرم افزارها طراحی و پیاده میشود و متصل به شبکه میباشد. مشابه فارسی این نرم افزار کم تر دیده شده است. با طراحی این نرم افزار سعی بر آسان تر کردن برنامه ریزی، زمانبندی و مدیریت پروژه های مختلف برای کاربران شده است.

### مسائل حقوقی :

- (۱) حفظ حریم شخصی و حقوقی افراد و شرکت ها، به طور مثال پاک شدن اطلاعات پروژه ها از سرور بعد از اتمام پروژه و حذف آن توسط مدیر پروژه
- (۲) رعایت مسائل مالی و حقوقی در رابطه با فروش نرم افزار در کافه بازار و یا فروش به افراد حقوقی



## ویژگی های اصلی

### موارد فنی :

#### (۱) نیازهای سیستمی

این اپلیکیشن بر روی device هایی که از سیستم عامل اندروید ۲ به بالا استفاده میکنند قابل اجرا و دسترسی خواهد بود .

اتصال امن و مطمئن به اینترنت برای ارسال و ذخیره ی داده ها مورد نیاز میباشد

این اپلیکیشن با زبان فارسی طراحی خواهد شد و device کاربر باید قابلیت پشتیبانی زبان فارسی را داشته باشد .

سیستم باید قابلیت ارتباط با یک میل سرور را نیز داشته باشد تا در صورت نیاز به صورت خودکار اطلاعات و notification هایی را برای کاربر خاصی ارسال نماید(این قابلیت در صورت داشتن زمان به ویژگی ها اضافه خواهد شد)

#### (۲) نیازهای مربوط به کارایی

از آنجا که تعداد کاربران شرکت کننده در یک پروژه زیاد میباشد سیستم باید قابلیت هماهنگی و مدیریت بالای اطلاعات را دارا باشد .

خطوط ارتباطی نیز علاوه بر امن بودن باید سریع و قابل اطمینان باشد.

### زمانبندی:

فاز ۱ – زمان تحویل ۹۳/۸/۱۰

- کلیت نرم افزار
- امکان عضو شدن افراد و ورود اعضا
- وجود پروفایل برای اعضا



## فاز ۲ - زمان تحویل ۹۳/۹/۱

- تعریف پروژه توسط مدیر پروژه
- امکان اضافه کردن اعضا توسط مدیر پروژه و امکان دادن قابلیت اضافه کردن اعضا به هر فرد توسط مدیر پروژه
- امکان تعریف task
- تعیین مهلت زمانی برای هر task توسط مدیر و fail شدن آن بعد از گذشت آن زمان
- امکان اطلاع رسانی به اعضای ارشد درباره اتمام یک task
- نمایش اعضای شرکت کننده در پروژه
- امکان شرکت فرد در چندین پروژه
- وجود بخش معرفی پروژه شامل توضیحات پروژه و میزان پیشرفت آن

## فاز ۳ - زمان تحویل ۹۳/۹/۲۴

- تیک خوردن یک task به عنوان done ، بعد از تایید توسط مدیر
- Notification برای بخش های مختلف
- نمودار دایره ای میزان پیشرفت پروژه
- امکان حذف پروژه ای که تمام شده
- امکان قرار دادن لینک پروژه برای پروژه های نرم افزاری
- امکان سرچ نام پروژه در بین پروژه های هر فرد

## فاز ۴ - زمان تحویل ۹۳/۱۰/۶

- امکان ارتباط با تهیه کنندگان نرم افزار و ارسال نظرات و پیشنهادات



## ابزار های مورد استفاده :

### (۱) Android Studio

همانطور که اشاره شد ، هدف نهایی این پروژه تولید نرم افزاری تحت پلت فرم اندروید می باشد ، یکی از چالش هایی که اعضای گروه با آن مواجه شدند انتخاب بین نرم افزار android studio و Eclipse می باشد ، که هر کدام نسبت به دیگری مزیت هایی دارند. در حال حاضر نرم افزار اندروید استودیو به دلایل زیر انتخاب شده است اما در صورت بروز مشکل ممکن است نرم افزار به Eclipse تغییر کند.

اندروید استودیو دارای :

- سیستم انعطاف پذیر بر پایه Gradle ( Gradle دارای قابلیت های وسیع می باشد)
  - ساخت انواع گونه های متفاوت نسل های APK
  - پشتیبانی از الگوهای گسترده برای خدمات گوگل و انواع مختلف دستگاه
  - ویرایشگر غنی Layout ها همراه با پشتیبانی از ویرایش قالب ها
  - ابزارهایی برای گرفتن عملکرد، قابلیت استفاده، نسخه های سازگار و مشکلات دیگر
- می باشد

از طرفی به مراتب android studio در قابلیت تکمیل کد بهتر و قوی تر است و نتایج دقیق تری را میدهد.

رابط کاربری android studio بسیار بهتر است.

از لحاظ محیط کاری و پروژه ها android studio به مراتب بهتر است.

اما مشکلات احتمالی در رابطه با این نرم افزار این است که نسخه ی بتا موجود می باشد و افراد کمی با این نرم افزار نسبت به Eclipse کار کرده اند.



## ۲) نرم افزار های design و طراحی UI

برای طراحی GUI نرم افزار پروژه ممکن است از نرم افزار های design و طراحی استفاده کنیم.

## ۳) Visual paradigm

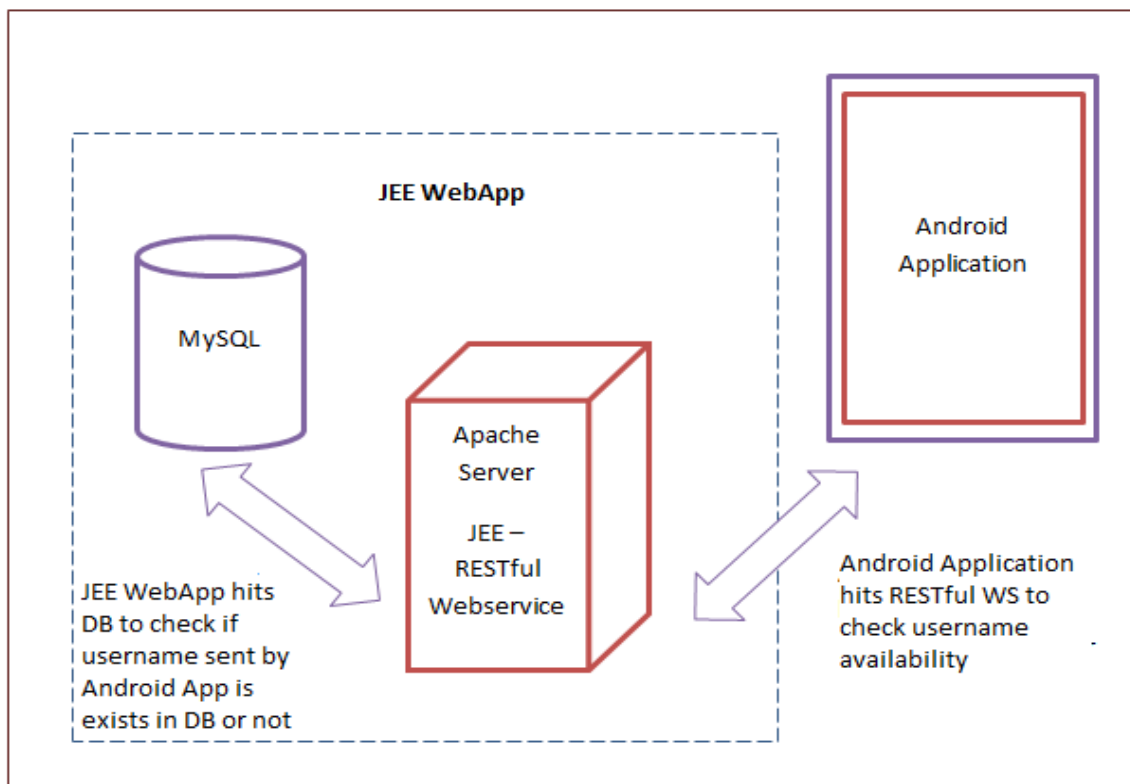
از ویژوال پارادایم برای طراحی جداول پایگاه داده و تولید کد مربوطه استفاده خواهیم کرد.

## ۴) MySQL

قسمت دیتا بیس مهمترین نقش در این نرم افزار را ایفا میکند ، برای کار با پایگاه داده از نرم افزار MySQL استفاده خواهیم کرد .

## ۵) RestFUL webservice

در ضمن در این پروژه یک وب سرویس فراهم خواهیم کرد. از جمله وب سرویس های موجود برای نرم افزار های اندروید soap و RESTFUL میباشد انتخاب ما برای این پروژه REST میباشد چون استفاده از آن به مراتب از soap آسان تر است .







## ۶ Trello و GitBash

برای انتقال اطلاعات بین اعضای گروه از Gitbash و برای مدیریت کلی پروژه و کنترل آن از نرم افزار Trello استفاده میکنیم .

### محدودیت ها :

با توجه به اینکه اطلاعات موجود در پروفایل هر فرد مربوط به کل پروژه هایی است که وی در آن شرکت داشته و همچنین شامل بخشی از داده های مربوط به فعالیت سایر اعضا میباشد پایگاه داده ی مورد استفاده باید سریع و قابل دسترس باشد و به راحتی با سرور در تعامل باشد زیرا تمام کارایی این برنامه به تعامل غیرحضوری افراد باهم، با اطمینان از صحت و کامل بودن داده ها بستگی خواهد داشت . در ضمن این پایگاه داده باید قابلیت پشتیبانی حجم بالای اطلاعات کاربران را داشته باشد.

### پیش فرض ها :

این برنامه تنها به یک سرور خارجی برای برقراری ارتباط و تبادل داده ها بین کاربران احتیاج خواهد داشت و به روز شدن اطلاعات نیاز به دسترسی به اینترنت دارد. سیستم به یک میل سرور متصل خواهد بود.

### متدولوژی :

متدولوژی موردنظر تیم ما Agile می باشد. در این متدولوژی حجم مستندسازی نسبت به سایر متدها به طور قابل توجهی کاهش می یابد و همچنین تکیه بیشتر بر روی کدنویسی است. متدولوژی Agile مجموعه روشهایی است که باعث می شود تا نرم افزار تولید شده کاملاً با نیازهای مشتریان مطابقت داشته باشد.

متدهای چابک وظایف را به گامهای کوچک با کمترین میزان برنامه ریزی می شکنند که به طور مستقیم با برنامه ریزی های طولانی مدت درگیر نیستند. تکرارها فریم های (بسته های زمانی) کوتاه مدتی هستند که معمولاً بین یک تا چهار هفته طول می کشند. هر تکرار دارای یک تیم متقابل



عملکردی در تمام مأموریت‌ها است : تحلیل نیازمندی‌ها، طراحی، کدنویسی، واحد تست، و قبولی در تست. در پایان هر تکرار یک محصول کاری به ذی‌نفعان نشان داده می‌شود. این، ریسک کلی را به حداقل رسانده و اجازه می‌دهد پروژه خیلی سریع با تغییرات منطبق شود.

یکی از تفاوت‌های بین چابک و آبشاری، این است که تست نرم‌افزار در نقاط مختلفی در چرخه‌ی عمر توسعه‌ی نرم‌افزار انجام می‌شود. در مدل آبشاری، یک فاز تست به صورت جداگانه بعد از پیاده‌سازی وجود دارد.

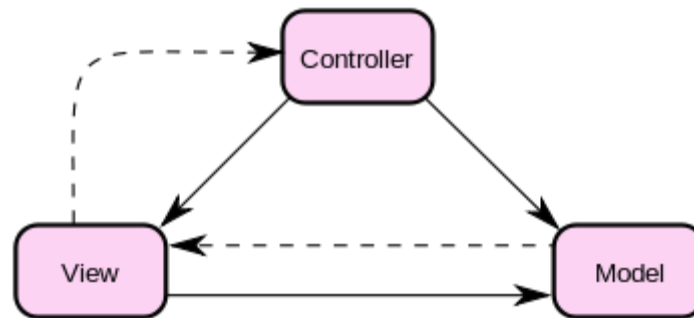
اصول ۱۲ گانه Agile عبارتند از :

۱. بالاترین اولویت در این متدولوژی جلب رضایت مشتری با تحویل زود هنگام نرم‌افزاری توانمند است .
۲. استقبال از تغییر نیازمندی‌ها، حتی در اواخر فرآیند توسعه .
۳. تحویل نرم‌افزار قابل استفاده با فاصله زمانی سه هفته یک بار و یا سه ماه یک بار.
۴. ذی‌نفعان و توسعه‌دهنده‌ها می‌بایست به صورت روزانه در طول پروژه با هم کار کنند.
۵. پروژه‌ها به دست افراد با انگیزه سپرده شود ، فضای لازم به آنها داده شود تا کارها را به درستی انجام دهند.
۶. کارآمدترین و موثرترین روش انتقال اطلاعات به تیم توسعه و تبادل آن در میان اعضای تیم ، گفتگوی چهره به چهره است.
۷. نرم‌افزار قابل استفاده اصلی ترین معیار سنجش پیشرفت است
۸. فرآیند های Agile توسعه پایدار را ترویج می‌دهند. حامیان مالی، توسعه‌دهندگان و کاربران باید بتوانند سرعت پیشرفت ثابتی را برای مدت نامحدودی حفظ کنند.
۹. توجه مداوم به برتری فنی و طراحی خوب باعث افزایش کیفیت تولید می‌شود.
۱۰. باید ساده ترین راه که با هدف پروژه سازگار است را انتخاب نمود و از گذاشتن وقت بر روی مشکلاتی که ممکن است در آینده رخ دهد صرف نظر کرد.
۱۱. بهترین مدیریت، تحلیل نیازمندی‌ها و طراحی از تیم‌های خود سازمان ده (هرفرد در تیم بر کل پروژه تاثیر دارد) پدید آور می‌شود.



۱۲. در فواصل منظم، تیم نشان میدهد که چگونه میتواند در تولید نرم افزار موثرتر باشد و سپس تیم رفتار خود را بر اساس بازتاب این تفکر تنظیم و هم سو می نماید.

### معماری مورد استفاده :



با توجه به موضوع و ویژگی های پروژه و نیازهای مربوط به پیاده سازی اپلیکیشن، بهتر دیدیم از معماری مدل-نما-کنترلر یا MVC (Model-View-Controller) استفاده کنیم.

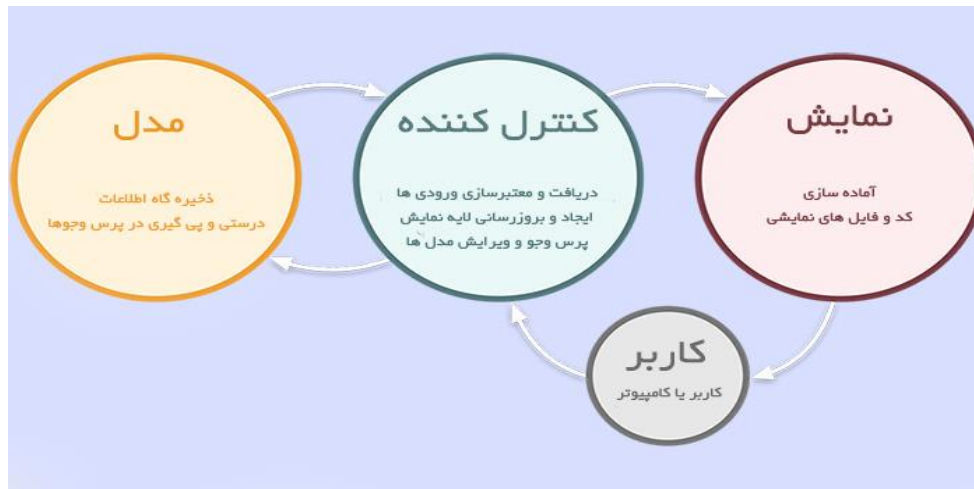
معماری MVC به نوعی از معماری در مبحث برنامه نویسی گفته می شود که تاکید بسیار به لایه بندی سیستم دارد. در واقع هدف از معماری نرم افزاری MVC ایجاد معماری چند لایه ای جهت جداسازی قسمت های مختلف برنامه است. به طور دقیق تر MVC باعث جدا شدن بخش های منطقی برنامه که شامل اطلاعات، سطح دسترسی ها، چک کردن صحت داده ها از لایه نمایش (presentation) یا در واقع همان لایه ای که مستقیماً با کاربر نهایی (end-User) در ارتباط است، می شود.

هدف الگوی معماری MVC صرفاً یکپارچگی در معماری نرم افزار است و به کمک آن بدست گیری نرم افزار در راستای مدیریت و توسعه به سهولت انجام می گیرد.

مزیت کلیدی روش MVC این است که اجزا نسبت به یکدیگر به اصطلاح loosely هستند. بدین معنا که هر قسمت مجزا از برنامه هدف خاص خود را دارد و می تواند بدون تاثیر بر روی دیگر قسمت ها به طور مستقل تغییر کند.



در ادامه بررسی معماری نرم افزاری MVC سه بخش اصلی در معماری نرم افزاری MVC را به طور مختصر توضیح می دهیم.



#### • مدل ( Model )

بار اصلی معماری MVC بر عهده این بخش است. مدل میتواند با داده ها در ارتباط باشد. الزاما منظور از داده حتما ارتباط با پایگاه های داده همچون MySQL و MSSQL نیست، حتی منبع داده ها در بخش Model میتواند یک آرایه از اعداد و یا هر چیز دیگری باشد. مدل وظیفه چک کردن داده ها جهت صحت درستی آنها را هم بر عهده دارد.

#### • نما ( View )

نما وظیفه ی برقراری ارتباط با کاربر نهایی و گرفتن داده از کاربر و نمایش داده های آماده به کاربر از طریق برقراری ارتباط با دو بخش دیگر یعنی Model و Controller را دارد. در واقع نکته مهمی که در بخش View باید آن را مد نظر داشت این است که این لایه مسئول کنترل صحت داده های وارد شده از طریق کاربر و همچنین مسئول صحت داده های نشان داده شده به کاربر نیست و در واقع این بخش با داده های خام کار می کند.

به عنوان یک مثال ساده، بسیاری از برنامه نویسان برای زمانی که در فرم Login برنامه، کاربر کلمه عبور خود را وارد می کند، در همان فرم Login اقدام به چک کردن پسورد مبنی بر صحت آن و دیگر موارد مورد نظرشان می کنند؛ که این عمل در معماری MVC قابل قبول نیست! در



واقع برای حل این مسئله در معماری MVC ، در فرم Login ، هنگامی که کاربر کلمه عبور را وارد کرد و دکمه Login را زد، کلمه عبور داده شده بدون هیچ گونه اعمالی مانند رمزگذاری (encrypt) کردن و ... به بخش های دیگر فرستاده می شود و فقط یک نتیجه ساده مبنی بر این که کاربر اجازه ورود دارد یا خیر را از بخش های دیگر دریافت می کند که بر اساس آن اجازه ورود کاربر به برنامه داده می شود.

### • کنترلر ( Controller )

در این لایه درخواست های کاربر پردازش می شود و به قسمت های مربوطه هدایت می شود. این لایه، واسطه ی ارتباط بین لایه ی نما و مدل است، بدین صورت که درخواست های کاربر از طریق این لایه به مدل ارسال می شود.

### روند اجرای برنامه در MVC

در معماری MVC روند کلی برنامه به این شکل است که کاربر تقاضای خود را از طریق واسط های برنامه نویسی نظیر Form ها، Link ها و ... از بخش View درخواست می کند. بخش View درخواست ها را به بخش Controller فرستاده و این بخش با برقراری ارتباط با بخش Model در خواست های کاربر را پردازش کرده و پس از پایان پردازش زمانی که خروجی درخواست داده شده آماده گردید، بخش Controller بخش View را آگاه می سازد تا خود را بر اساس تغییرات جدید به روز سازد.

### نحوه تست و انواع تست نرم افزار

تست نرم افزار را در چهار سطح مختلف انجام می دهیم که این چهار مرحله به صورت ترتیبی انجام می پذیرند و عبارتند از :

### • تست واحد (Unit testing)

هدف در این سطح از تست این است که آیا هر واحد مورد نظر که یکی از اعضای گروه طراحی کرده است به تنهایی کار خود انجام می دهد یا نه. در این حالت چند نوع تست انجام می دهیم :



۱. بررسی سطر به سطر کد (Code coverage) در این حالت باید سیستم را به گونه ای اجراء و بررسی کنیم که مطمئن شویم سطر به سطر کد برنامه حداقل یکبار اجراء شده است.

۲. بررسی همه انشعابها در کد برنامه

۳. بررسی همه حلقهها در برنامه

۴. مدیریت خطای مطلوب : بررسی اینکه اگر به یک متد یک ورودی نامعتبر (مثلا برای پسورد) وارد شود، نحوه آگاه سازی و نمایش مطلوب خطا برای کاربر چگونه باشد.

### • تست مجتمع سازی (Integration Testing)

ما نیاز داریم تا مطمئن شویم واحدها در کنار هم به درستی فراخوانی می شوند و داده های درستی را در زمان مناسبی از طریق واسط های آنها عبور می دهند. تست مجتمع سازی یکی از مهمترین و شاید مهمترین سطح از تست باشد.

برای اینکار آزمایش کننده نرم افزار به مستندات نرم افزار مراجعه می کند تا مشخص کند که سیستم در مقابل یک عمل خاص چه پاسخی را باید بدهد. سپس داده های را برای هر کدام از عملیات انتخاب می کند و رفتار سیستم را در مقابل آن داده ها با رفتار واقعی سیستم که در مستندات وجود دارد مقایسه و بررسی می کند :

۱. بررسی اینکه سیستم نیازمندیهای عملیاتی و غیر عملیاتی را تامین می کند یا نه.

۲. بررسی خروجی های سیستم، بررسی نوتیفیکشن های سیستم، اطلاع رسانی های مختلف در زمان موردنظر

۳. بررسی رفتار سیستم در برابر پردازش چندین پروژه در حال اجرا و پرس و جویهای بزرگ و سنگین



## • تست سیستم (System Testing)

نرم افزار ما خود جزئی از یک سیستم است و نیاز است که با عناصر دیگر این سیستم مانند سخت افزارها ارتباط برقرار کند و با آنها مجتمع شود. پس نیاز داریم تا مطمئن شویم که سیستم به عنوان یک واحد به طور کامل عمل خواهد کرد و نیازمندیهای سیستم را برآورده می کند. این سطح از تست آخرین سطحی است که توسط توسعه دهندگان تا قبل از تحویل نرم افزار به کاربر نهایی برای تست عملکرد آن صورت می گیرد. در سطح مهندسی سیستم از تست جعبه سیاه استفاده می شود. برای نمونه موارد زیر در تست سیستم مورد بررسی قرار می گیرد:

۱. تست امنیت (Security Testing): برای اطمینان از عملکرد سیستم در برابر نفوذگران ما باید مکانیزم امنیتی در سیستم داشته باشیم تا مطمئن شویم که سیستم می تواند نفوذهای غیر قانونی را تشخیص دهد و در برابر آنها عکس العمل نشان دهد.
۲. تست بازیابی (Recovery Testing): در این نوع آزمایش باعث ایجاد مشکل و از کار افتادن سیستم با ورودیهای غیرمنتظره و اشتباه می شویم و بررسی می کنیم که آیا سیستم می تواند خود را به طور خودکار بازیابی کند و به فعالیت خود ادامه دهد.

## • تست پذیرش (Acceptance Testing)

سپس ما به آزمایشی نیاز داریم که توسط کاربران نهایی، مشتریان و نه توسعه دهندگان صورت می گیرد و هدف آن است که کاربر مشخص کند عملیاتی که برنامه انجام می دهد نیازمندیهای آنها را برآورده می کند یا نه. تست پذیرش دارای انواع مختلفی است که می توان به موارد زیر اشاره کرد:

### ۱. آلفا (Alpha)

نسخه آلفای یک نرم افزار، برای آزمایش کنندگان نرم افزار - که اشخاصی غیر از مهندسين نرم افزار هستند - تهیه می گردد. تمام ویژگیها برای تمام سطوح دسترسی کنترل میشود و در صورت عدم وجود اشکال محصول از این تست عبور میکند. تست آلفا میتواند شامل تستهای functional و non-functional باشد.



منظور از تستهای functional بررسی عملکرد تمامی ویژگیها و توابع نرم افزار مدیریت پروژه میباشد و منظور از تستهای non-functional نیز بررسی ویژگیهایی مانند کارایی و محدودیتهای سیستم است.

## ۲. بتا (Beta)

نسخه بتا اولین نسخه ارایه شده در خارج از مجموعه تولید کننده نرم افزار است. نرم افزار سطح بتا همه ویژگیهای نسخه اصلی را داراست، اما امکان دارد اشکالات بسیار کمی داشته باشد. مشتریان فعلی، نرم افزار را به طور رایگان یا با قیمت کم دریافت می کنند. آزمایش کنندگان تعدادی از اشکالاتی را می یابند و گاهی ویژگیهای فرعی دیگری را که تمایل دارند در نسخه نهایی ببینند گزارش می دهند.

در نهایت پس از انجام تست های مختلف مرحله ای که همه اشکالات مهم برطرف گردیده است مرحله stable نرم افزار است که آماده ورود به بازار میباشد.

## نحوه نصب نرم افزار :

آسان ترین راه دانلود نرم افزار از بازار اندروید میباشد و یا فایل apk نرم افزار را از تهیه کنندگان دریافت کرده و نصب میکنید.





## افراد و نقش ها

مدیر پروژه : سارا متین فر

کلیه وظایف از قبیل تحلیل، کد نویسی، تست نرم افزار و ... مابین اعضای گروه تقسیم می شود و فردی به طور تخصصی روی یک بخش فعالیت نمیکند. درواقع در هر مرحله کلیه اعضا مشارکت دارند.

## پشتیبانی

برای پشتیبانی نرم افزار، در صورت وقوع هر خطایی در هنگام کار با آن، گزارش خطا به تهیه کنندگان اطلاع داده می شود تا هرچه سریعتر در جهت رفع مشکل اقدام کنند. همچنین قسمتی جداگانه جهت ارتباط با تهیه کنندگان و بیان نظرات و پیشنهادات در نظر گرفته شده و این نظرات از طریق پست الکترونیکی به آنها اطلاع داده خواهد شد.