

Workflow Checklist Package for Software System Development

N. Fareghzadeh*,¹

¹ Computer Science Department, Islamic Azad University (IAU), Khodabandeh Branch, Zanjan, Iran

ABSTRACT


Received: 15 February 2023

Accepted: 24 May 2023

KEYWORDS:

Check List,
Workflow,
Software Life Cycle,
Project Management,
Evaluation Metric,
System Development,

¹ Corresponding author

 n.fareghzadeh@iauz.ac.ir

In today's technology-oriented world, a wide range of commercial and administrative services are provided through the use of software systems. Due to the importance of a comprehensive and structured process in the development of a software systems and in this regard, in this paper proposes a useful management checklist cluster, called software system development workflow checklist. The proposed checklist is actually a package containing a subset of more detailed monitoring checklists in the software system development process. The suggested checklist is a tool to create a comprehensive management routine to evaluate the performance of tasks and structured workflow in the development of software systems. In addition to explaining the system workflow and clarifying the monitoring of software system development, the proposed workflow checklist package is proposed as a resource for working team members to evaluate and resolve differences of opinion and align executive tasks and also to avoid and prevent common management mistakes. It should be noted that based on the proposed checklist package, in the software system development workflow management process, project managers could monitor the system development with both a top-down and middle-out approach. This means that first, with a top-down view of each phase of the software life cycle, could receive the progress report and with the middle-out management view, evaluate and control items of the proposed checklist package. From our point of view, having such a workflow checklist package is one of the most essential activities in order to facilitate the management and auditing of the software development life cycle, which is needed for the integrated implementation of organizational projects and their structured and comprehensive management.



NUMBER OF REFERENCES

29



NUMBER OF FIGURES

1



NUMBER OF TABLES

0

فصلنامه تخصصی آرمان پردازش (APJ)

Homepage: www.armanprocessjournal.ir



پکیج چک لیست گردش کار برای توسعه سیستم های نرم افزاری

نویسنده: **فارغ زاده***^۱

^۱ دانشکده علوم کامپیوتری، دانشگاه آزاد اسلامی، واحد خدابنده، زنجان، ایران

چکیده

در دنیای تکنولوژی محور امروزی طیف وسیعی از خدمات کاربردهای تجاری و اداری از طریق بهره گیری از سیستم های نرم افزاری صورت می پذیرد. به دلیل اهمیت وجود یک فرایند جامع و ساخت یافته در توسعه و ارزیابی یک سیستم نرم افزاری و در این راستا، در این مقاله قصد داریم مجموعه چک لیست مدیریتی به نام بسته چک لیست گردش کار توسعه سیستم نرم افزاری پیشنهاد نمائیم. این چک لیست در واقع یک پکیج حاوی زیرمجموعه ای از چک لیست های جزئی تر پایشی و نظارتی در فرایند توسعه سیستم نرم افزاری می باشد. پکیج چک لیست پیشنهادی ابزاری در جهت ایجاد روال مدیریتی منظم جهت پایش و ارزیابی انجام وظایف و بنیان گذاری روال گردش کار ساخت یافته در توسعه سیستم های نرم افزاری می باشد. پکیج چک لیست گردش کار پیشنهادی علاوه بر تبیین و رصد جزئیات گردش کار سیستم و شفاف سازی نظارت ها بر توسعه سیستم نرم افزاری، به عنوان منبعی برای اعضای تیم های کاری در جهت حل و رفع اختلاف نظرات و همسوسازی اقدامات اجرایی تی های کاری مطرح می شود و همچنین برای پرهیز و جلوگیری از اشتباهات رایج مدیریتی قابل بکارگیری می باشد. لازم به ذکر است بر اساس پکیج چک لیست پیشنهادی، در بخش مدیریت گردش کار توسعه سیستم نرم افزاری، مدیر سیستم هم با دیدگاه بالا به پایین و هم دیدگاه میانی توسعه سیستم را پایش و نظارت می نماید. بدین معنی که ابتدا با دیدگاه بالا به پایین از هر فاز چرخه حیات نرم افزار گزارش تایید انجام وظایف را دریافت نموده و سپس با دیدگاه مدیریتی میانی آیتم های چک لیست های کلیدی پیشنهادی را ارزیابی و کنترل می نماید. از دیدگاه ما، داشتن چنین چک لیست گردش کاری یکی از ضروری ترین فعالیت ها در راستای تسهیل مدیریت و نظارت بر چرخه حیات توسعه سیستم های نرم افزاری است که برای پیاده سازی اصولی پروژه های سازمانی و مدیریت ساخت یافته و جامع آنها آن مورد نیاز می باشد.

واژگان کلیدی:

چک لیست،
گردش کار،
چرخه حیات نرم افزار،
مدیریت پروژه،
معیار ارزیابی،
توسعه سیستم،


تعداد مراجع
۲۹


تعداد شکل ها
۱

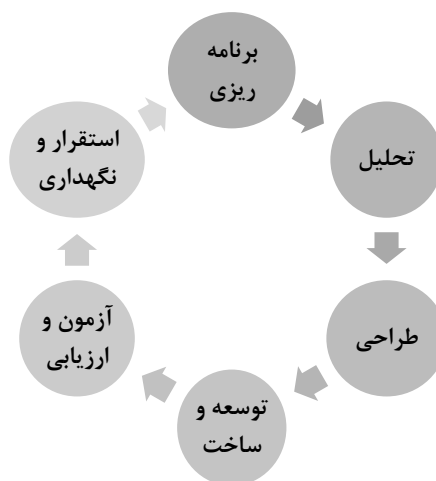

تعداد جداول
۰

مقدمه

اطمینان و با صرفه بوده و بر روی سیستم‌ها به‌طور کارآمدی عمل کنند، می‌باشد. در واقع مهندسی نرم‌افزار یک روش سیستماتیک، منظم و دقیق برای ساخت و ارائه محصول نرم‌افزاری با کیفیت است [۲]. چرخه حیات توسعه نرم‌افزار^۱، کاربرد روش‌های استاندارد کسب و کار در ساخت برنامه‌های نرم‌افزاری است. چرخه حیات توسعه نرم‌افزار می‌تواند اقدامات اجرایی فازها، قوانین، محدوده‌ها و تاخیرات را پیش‌بینی نماید تا توسعه‌دهندگان بتوانند سایر وظایف را آسان‌تر و به شکل حرفه‌ای‌تر بر عهده بگیرند. چرخه حیات توسعه نرم‌افزار نشان می‌دهد که در فرایند توسعه سیستم نرم‌افزاری چه اتفاقی می‌افتد و دقیقاً در کجای روند توسعه بهبودهایی می‌تواند انجام شود. مانند بسیاری از فرآیندهای کسب و کار، چرخه حیات توسعه نرم‌افزار با هدف تجزیه و تحلیل و بهبود روند ایجاد نرم‌افزار انجام می‌شود. این امر یک دیدگاه مقیاس‌پذیر از پروژه، از کدگذاری روزانه گرفته تا مدیریت تاریخ تولید را ایجاد می‌کند. این چرخه به‌طور معمول به شش تا هشت مرحله تقسیم می‌شود که مهم‌ترین این مراحل عبارتند از: برنامه‌ریزی، تحلیل، طراحی، توسعه و ساخت، آزمون و ارزیابی، استقرار و نگهداری [۳]. شکل زیر شمای کلی از مراحل توسعه سیستم نرم‌افزاری را نمایش می‌دهد:

با گسترش تکنولوژی‌های جدید بویژه نفوذ کارکردهای فناوری اطلاعات و ارتباطات در سراسر جهان و همچنین جهانی‌شدن پروژه‌های سازمانی، مکانیزه سازی کاربردهای مختلف در قالب پروژه‌ها و کاربردهای نرم‌افزاری اهمیت ویژه‌ای یافته است. اساساً نرم‌افزار، مجموعه‌ای از برنامه‌های رایانه‌ای، رویه‌ها و مستندات است که انجام کاربردها و وظائف مرتبط بر روی یک سیستم رایانه‌ای را بر عهده دارد. نرم‌افزارها محصولات هستند که توسط شرکتهای نرم‌افزاری، برنامه‌نویسان، و توسعه‌دهندگان نرم‌افزارهای آزاد جهت انجام هدفی خاص طراحی و ساخته می‌شوند. نرم‌افزارها از منظر ابعاد مختلف قابل رده‌بندی می‌باشند. امروزه میلیون‌ها شرکت نرم‌افزاری در سرتاسر جهان فعالیت می‌کنند و هرکدام محصولات خود را جهت برآورده نمودن نیازهای مشتریان خود در زمینه‌های گوناگون از قبیل حسابداری، انبارداری، فروشگاه‌ها، مدیریت، بایگانی، سندداری و ... ارائه می‌کنند. کاربران مختلف بر اساس نیاز، نرم‌افزار کاربردی مناسب خود را انتخاب می‌کنند و از امکانات آن بهره‌مند می‌شوند [۱].

علم مهندسی نرم‌افزار روال بهره‌گیری و استفاده از اصول مهندسی مناسب در جهت تولید و ارائه محصولات نرم‌افزاری با کیفیت که قابل



شکل ۱. شمای عمومی چرخه حیات توسعه سیستم نرم‌افزار

دارد. برنامه‌ریزی اصولاً باید محدوده و هدف برنامه را به وضوح مشخص نماید. این برنامه دوره و مفاد تیم را برای ایجاد موثر نرم‌افزار ترسیم کرده و همچنین محدودیت‌هایی را برای کمک به جلوگیری و خارج نشدن پروژه از مسیر اصلی آن تعیین می‌کند. نیازمندی‌های نرم‌افزار^۲ بخشی از برنامه‌ریزی در نظر گرفته می‌شود تا مشخص شود نرم‌افزار قرار است دقیقاً چه کاری انجام دهد و الزامات آن تعیین گردند. نیازمندی‌ها همچنین شامل تعریف منابع مورد نیاز برای ساخت پروژه است [۴]. فاز طراحی روش کار نرم‌افزار را مدل‌سازی می‌کند. برخی از مهم‌ترین جنبه‌های این فاز طراحی معماری، ساختار داده، مشخصه

لازم به ذکر است، مدیران پروژه‌های نرم‌افزاری گاهی و بنا به سبک توسعه پروژه می‌توانند مراحل فوق را با فازها و مراحل دیگر ادغام نمایند و یا برخی گام‌های موردنظر را به چرخه فوق بیفزایند. در مرحله برنامه‌ریزی، رهبران پروژه شرایط پروژه را ارزیابی می‌کنند. این مرحله شامل محاسبه هزینه‌های کارگری و مادی، ایجاد یک جدول زمانی با اهداف و ایجاد تیم‌های پروژه و ساختار رهبری است. برنامه‌ریزی می‌تواند شامل بازخورد ذینفعان نیز باشد. ذینفع هر شخصی است که می‌خواهد از این برنامه بهره‌مند شود. در این گام دریافت بازخورد از مشتریان بالقوه، توسعه‌دهندگان، کارشناسان موضوع و نمایندگان فروش اهمیت

² Software Requirements

¹ Software Life Cycle (SLC)

این راستا، در این مقاله قصد داریم مجموعه چک لیست های اساسی و مهمی را که می توانند در توسعه موثر و هدفمند سیستم های نرم افزاری الهام بخش باشند پیشنهاد نمائیم.

پکیج چک لیست گردش کار توسعه سیستم نرم افزاری

بصورت عمومی یک چک لیست^۱ مجموعه‌ای از آیتم های یادآوری است که برای مدیریت کارهای تکراری و فرایندهای مداوم ایجاد می شود. در چک لیست معمولاً گام‌های انجام فعالیت ها به ترتیب اولویت نوشته می‌شوند و کاربران پس از انجام هر کدام از آن‌ها روبه‌رویشان علامت تیک می‌زنند و به سراغ گام بعدی می‌رود. بسیاری از افراد فهرست کارها را با چک لیست یکی می‌دانند. در حالی که این دو به طور کامل با یکدیگر فرق دارند. فرق اصلی فهرست کارها با چک لیست در این است که عموماً در فهرست کارها به سوال چه کارهایی (What) پاسخ می‌دهیم ولی در یک چک لیست عموماً به سوال چگونه انجام کار (How) پاسخ می‌دهیم [۹-۱۰].

امروزه با توجه به توسعه روزافزون صنعت نرم افزار، استفاده از روال های استاندارد برای ایجاد هماهنگی میان اعضای تیم های درگیر در توسعه پروژه های نرم افزاری و اطمینان خاطر یافتن از اجرای درست و به موقع فرایندهای مرتبط و تکمیل اصولی فازهای چرخه حیات نرم افزار اجتناب ناپذیر است. به دلیل اهمیت وجود یک فرایند جامع و ساخت یافته در مدیریت توسعه یک سیستم نرم افزاری و در این راستا، در این مقاله قصد داریم مجموعه چک لیست های مدیریت پروژه نرم افزاری اساسی را تحت عنوان بسته چک لیست گردش کار توسعه سیستم نرم افزاری (SDWCP)^۲ پیشنهاد نمائیم. این چک لیست در واقع یک بسته حاوی زیرمجموعه ای از چک لیست های جزئی تر نظارتی در فرایند توسعه سیستم نرم افزاری می باشد. پکیج چک لیست پیشنهادی ابزاری در جهت ایجاد روال مدیریتی منظم جهت ارزیابی انجام وظایف و گردش کار ساخت یافته در توسعه سیستم های نرم افزاری می باشد. این پکیج چک لیست یادآوری می نماید که وظایف و فرایندهای کلیدی به علت فراموشی انجام آنها، ناتمام نمی مانند. پکیج چک لیست گردش کار پیشنهادی علاوه بر تبیین کار و مشخص نمودن چگونگی اجرای اصولی نظارت بر توسعه سیستم نرم افزاری، به عنوان منبعی برای اعضای تیم های کاری و در جهت حل و رفع اختلاف نظرات و همسوسازی عملکردها و اقدامات اجرایی مطرح می شود و همچنین برای پرهیز و جلوگیری از اشتباهات رایج مدیریتی قابل بکارگیری می باشد. استفاده از پکیج چک لیست گردش کار پیشنهادی می تواند مزایای متعددی داشته باشد که از جمله مهم ترین آنها می توان به موارد زیر اشاره کرد:

- تخصیص بهتر وظایف و مسئولیت ها به فازهای چرخه حیات نرم افزار و تیم های کاری مرتبط
- استانداردسازی جریان های کاری سیستمی

های فنی، ارتباطی و امنیتی می باشد. توسعه برنامه شامل نوشتن واقعی کدمنای برنامه ها است. روند کدگذاری شامل بسیاری از وظایف و کارهای دیگر است. توسعه دهندگان باید مهارت‌های خود را افزایش دهند یا به صورت تیمی کار کنند. یافتن و رفع خطاها و اشکالات بسیار حیاتی است. برخی وظایف اغلب روند توسعه را متوقف می‌کنند، مانند انتظار برای نتایج آزمون یا فشرده‌سازی کد برای اجرای یک برنامه. فاز آزمون یک برنامه قبل از در دسترس قرار دادن آن برای کاربران بسیار مهم است. بسیاری از آزمایشات می‌توانند خودکار باشند، مانند آزمایشات امنیتی. آزمون باید کارکرد صحیح هر عملکرد را تضمین کند. قسمت‌های مختلف برنامه نیز باید آزمایش شوند تا به طور یکپارچه با هم کار کنند؛ آزمایش عملکرد، برای کاهش تاخیر در پردازش انجام می‌گردد. مرحله آزمایش به کاهش تعداد اشکالاتی که کاربران با آن روبرو می‌شوند کمک می‌کند. این امر منجر به رضایت بیشتر کاربر و میزان استفاده بهتر می‌شود [۵]. در فاز نهایی چرخه حیات نرم افزار که فاز استقرار و نگهداری نام دارد، برای انتقال سیستم آماده‌شده به محیط مشتری، تدابیر لازم اتخاذ می‌شود. در طول این فاز، برای اطمینان از بی‌نقص بودن و تحویل کامل نرم‌افزار به مشتری، آخرین بررسی‌های لازم روی نرم‌افزار انجام می‌شود و پس از انتقال نرم افزار به محیط مقصد، پشتیبانی، رفع نقص، نگهداشت و آموزش کارکردهای نرم افزار به کاربران صورت می‌پذیرد [۶].

لازم به ذکر است، در فرایند ارائه محصول نرم افزاری با کیفیت، نگه داشتن هزینه در محدوده بودجه مشتری و تحویل پروژه طبق برنامه ریزی انجام شده، بخش اساسی سازمان دهی و کنترل پروژه های نرم افزاری است. عوامل مختلف داخلی و خارجی وجود دارد که ممکن است این مثلث محدود کننده سه گانه را تحت تأثیر قرار دهد. هر یک از سه عامل می تواند به شدت بر روی دو عامل دیگر تأثیر بگذارد. بنابراین، مدیریت موثر پروژه نرم افزاری برای ترکیب نیازهای کاربر به همراه محدودیت های بودجه و زمان ضروری است [۷]. مدیر پروژه نرم افزاری شخصی است که مسئولیت اجرای پروژه نرم افزاری را بر عهده می‌گیرد. مدیر پروژه ممکن است هرگز مستقیماً در تولید محصول نهایی دخیل نباشد، اما فعالیت های مربوط به تولید را کنترل و مدیریت می‌کند. یک مدیر پروژه از نزدیک بر روند توسعه نظارت می‌کند، برنامه های متنوعی را تهیه و اجرا می‌کند، منابع لازم و کافی را ترتیب می‌دهد، ارتباط بین همه اعضای تیم را حفظ می‌کند تا به مسائل مربوط به هزینه، بودجه، منابع، زمان، کیفیت و رضایت مشتری منجر شود [۸]. مدیران پروژه های نرم افزاری برای سازماندهی اصولی وظایف و گردش کار در پروژه های نرم افزاری نیازمند بهره گیری از یک روال منظم می باشند. چک لیست ها در این شرایط بسیار کمک کننده هستند و بمنظور پرهیز از خطاهای ارزیابی و اشتباهات مدیریتی موثر می باشند. لذا بنا به اهمیت مدیریت ساخت یافته در توسعه یک سیستم نرم افزاری و در

³ Software Development Workflow Checklist (SDWC)

¹ Check List

² To-Do List

- حرکت بسوی جامعیت و ایجاد زبان مشترک جهت تعامل بین تیم های کاری درگیر پروژه
- افزایش کیفیت انجام کار و تکمیل بهینه تر فرایندها و کارها
- شفاف سازی و تاکید بر نقاط بحرانی پروژه
- افزایش ایمنی در کشف انحرافات و خطاها
- افزایش بهره وری فرایندها، خلاقیت و انگیزه در پیگیری گردش کار سیستم
- ارزیابی بهتر خطوط مبنا، کنترل ها
- ارزیابی بهتر مسیر بحرانی سیستم و کارهای موردی و اضطراری
- مدیریت موثر کیفیت سیستم و فرایندهای مرتبط
- حذف دوباره کاری ها و پرهیز از اتلاف زمان
- ایجاد ابزاری مشورتی برای اطمینان از نگاه همه جانبه به فعالیت های پروژه

چک لیست گردش کار پیشنهادی به مدیران پروژه و اعضاء تیم های کاری درگیر توسعه سیستم نرم افزاری کمک می کند تا همواره و همچنان متمر ثمر و موثر باشند. این چک لیست معیاری برای جهت نمائی هدفمند و اصولی توسعه سیستم نرم افزاری فراهم می کند. این رویکرد درواقع روشی است که نشان می دهد که در حوزه مدیریت توسعه سیستم نرم افزاری چه محک ها و ارزیابی هائی باید انجام شود و به بیان ساده اجازه می دهد کنترل ابعاد مختلف انجام کار در پروژه را به دست گرفت. از دیدگاه ما، داشتن چنین چک لیست گردش کاری یکی از ضروری ترین فعالیت ها در راستای تسهیل مدیریت و نظارت بر چرخه حیات توسعه سیستم های نرم افزاری است که برای پیاده سازی اصولی پروژه های سازمانی و مدیریت آن مورد نیاز می باشد. لازم به ذکر است براساس چک لیست پیشنهادی، در بخش مدیریت گردش کار توسعه سیستم نرم افزاری، مدیر سیستم هم با دیدگاه بالا به پایین و هم دیدگاه میانی توسعه سیستم را نظارت می نماید. دیدگاههای بالا به پایین، پائین به بالا و دیدگاه میانی دیدگاههای رایج مدیریت پروژه می باشند که همواره مورد توجه بسیاری از تحقیقات مرتبط بوده اند [۱۴-۱۱]. در SDWCP ابتدا با دیدگاه بالا به پایین^۱ از هر فاز چرخه حیات نرم افزار گزارش پیشرفت انجام وظائف دریافت می گردد و سپس با دیدگاه مدیریتی میانی^۲ آیتم های چک لیست های کلیدی سیستم را ارزیابی و کنترل می گردد. چک لیست های کلیدی توسعه سیستم که درون SDWCP بکارگرفته می شوند در ادامه مقاله ذکر گردیده اند:

- چک لیست تدارکات و استراتژی سازی
- چک لیست تسطیح جریان کار
- چک لیست مدیریت منابع و ریسک ها
- چک لیست پایش و نظارت
- چک لیست ارزیابی کیفی

- چک لیست کنترل یکپارچگی
- چک لیست مستندات

همانطور که در قسمت های قبلی مقاله ذکر گردید، پکیج چک لیست پیشنهادی موازی با فازهای چرخه حیات سیستم می تواند با بهره گیری از کنترل ها و نظارت های دقیق و پایش جریان های کاری در سیستم به توسعه استاندارد و اصولی سیستم نرم افزاری بینجامد. در ادامه مقاله حاضر به شرح جزئیات چک لیست های SDWCP می پردازیم.

چک لیست تدارکات و استراتژی سازی

چک لیست تدارکات و استراتژی سازی در آغاز فرایند نظارت پروژه و با هدف تایید طرح ریزی و مقدمات و تدارک پیش نیازها در پیشبرد جریان کار و اجرای فازهای چرخه حیات نرم افزار و توسعه سیستم بکار گرفته می شود. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان اولین زیرمجموعه در پکیج SDWCP استفاده نمود:

- تبیین اهداف خرد و کلان سیستم و فرضیات
- شناسایی الزامات کمی و کیفی و تدارکات اولیه
- شناسایی ذینفعان پروژه و تعیین استراتژی مدیریت نیازمندی های ذینفعان
- تحلیل نیازمندی ها و محدودیت ها
- تعیین و تایید استراتژی و متدولوژی مناسب جهت توسعه نرم افزار و چرخه حیات سیستم
- چیدمان تیم های کاری و سرپرستان، تعیین مسئولیت ها و حدود اختیارات اعضاء
- شناسایی زیرسیستم های رسمی و الگو و فرهنگ تعاملی موجود در سازمان
- دریافت فرآیندها، رویه ها و اطلاعات روال های کاری در پروژه های قبلی مرتبط
- برنامه ریزی تعیین وظائف هر فاز در چرخه حیات نرم افزار

چک لیست تسطیح جریان کار

چک لیست تسطیح جریان کار به چک لیست دیگری در پکیج SDWCP و با هدف تسطیح، ردیابی و پایش اقدامات اجرایی مولد جریان های کاری سیستم که در فازهای مختلف چرخه حیات نرم افزار انجام می شوند، مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود:

- تعیین مسیر بحرانی و خطوط مبنای پروژه
- تکمیل گستره اقدامات پایه توسعه محصول نرم افزاری (تولید / تهیه / ایجاد و تغییر اقلام تحویلی)

² Middle-Out Approach

¹ Up-Down Approach

- تبیین ورودی ها و خروجی های هر فاز چرخه حیات
- سیستم نرم افزاری
- تفکیک و تسطیح کارکردهای جریان های کاری مختلف در چرخه حیات نرم افزار
- مدیریت تعاملات اجرائی و ارتباطی بین فازهای اجرائی
- تبیین، رده بندی و تسطیح اقدامات اجرائی مرتبط
- تبیین خطوط مبنا و معیارهای ارزیابی
- برگزاری جلسات تشریح جزئیات عملیاتی اقدامات اجرائی با تیم های کاری
- تعامل و توافق با ذینفعان درخصوص اقدامات اجرائی
- درخواست و اعمال تغییرات و پیکربندی ها
- تعیین استاندارد ها، فرآیندها و شاخص های ارزیابی کیفی سیستم و زیرسیستم ها

چک لیست مدیریت منابع و ریسک ها

- اساسا موفقیت یک پروژه نرم افزاری تا حد زیادی به میزان ریسکی بستگی دارد که در آن پروژه وجود دارد. ریسک در مدیریت پروژه به هر رویداد غیرمنتظره ای گفته می شود که می تواند روی توسعه سیستم تأثیرگذار باشد [۱۵]. ریسک ها می توانند بر هر بخش یا قسمتی از پروژه مانند منابع انسانی موجود، فرایندها، فناوری و منابع در دسترس تأثیر بگذارند. درحالت کلی رویدادهای ریسک در یک پروژه می توانند مثبت یا منفی باشند. برای کمک به تمایز بین ریسک های مثبت و منفی در پروژه، برخی از افراد ترجیح می دهند ریسک های مثبت را فرصت بهبود توسعه سیستم و ریسک های منفی را تهدید این امر بدانند [۱۶]. بهره برداری از یک ریسک مثبت، یعنی انجام روش هایی که به افزایش احتمال وقوع آن کمک می کند و با بهره گیری از منابع موجود به شکل موثرتری پروژه را به اهدافش نزدیک تر می نماید. افزایش این ریسک به معنای تلاش برای افزایش فرصت های مفید در جهت بهبود توسعه سیستم است. درمقابل، ریسک های منفی پروژه تهدیدی برای موفقیت پروژه هستند و مدیریت ریسک های منفی، شامل اجتناب، انتقال یا کاهش ریسک خواهد بود. به عنوان یک مدیر پروژه، نباید به آگاهی از ریسک های پروژه اکتفا کرده، بلکه برای دستیابی به یک نتیجه مطلوب باید تمام منابع و ریسک های اصلی پروژه را شناسایی، ارزیابی، اولویت بندی و مدیریت کرد [۱۷]. چک لیست مدیریت منابع و ریسک ها به عنوان چک لیست بعدی در پکیج SDWCP و با هدف مدیریت موثر منابع و ریسک های محتمل در توسعه سیستم نرم افزاری، مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود [۱۹-۱۸]:
 - تخمین منابع مالی، بودجه، چارچوب های زمانی، منابع انسانی و تجهیزاتی پروژه
 - تخمین هزینه های آشکار و پنهان و سربارها در پروژه
- تعیین چارچوب های زمانی در مسیر بحرانی پروژه
 - شناسایی و اولویت بندی ریسک های مثبت و منفی پروژه
 - تحلیل کمی و کیفی ریسک ها و برنامه ریزی برای مواجهه با ریسک ها براساس اولویت
 - تبیین برنامه زمانی مبنا براساس رخدادهای کلیدی
 - تهیه برنامه زمانی جریان های کاری پروژه براساس مهلت انجام کار و ریسک های زمانی پروژه
 - ترسیم شبکه فعالیت های بنیادین پروژه براساس نیازمندی ها، محدوده های زمانی و محدوده مجاز تاخیر
 - برنامه ریزی اقدامات اجرائی مطابق برنامه و تامین نقدینگی پروژه، با توجه به ملاحظات درآمدی / تامین مالی پروژه
 - بهره گیری از دو رویکرد ((بالا به پایین)) و ((پایین به بالا)) برای تنظیم بودجه کلان و تفضیلی پروژه
 - تبیین بودجه حساب های کنترلی کارهای مستقل، فعالیت های وابسته و اقدامات پشتیبانی و موارد غیرمنتظره
 - مقایسه بودجه هزینه ای با بودجه درآمدی اقدام درجهت تعدیلات موردنیاز و متوازن سازی ریسک ها
 - مقایسه و تنظیم بودجه نقدی مصارف و منابع
 - تدقیق و متعادل سازی بودجه و هزینه ها براساس خروجی های چک لیست مدیریت منابع و ریسک ها
- ### چک لیست پایش و نظارت
- چک لیست پایش و نظارت به عنوان چک لیست دیگر در پکیج SDWCP و با هدف ردیابی و پایش اقدامات انجام شده و کنترل تغییرات درخواستی براساس تحلیل منابع و ریسک ها، مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود [۲۱-۲۰]:
 - تهیه برنامه نهایی پایش و نظارت فعال و غیرانفعالی سیستم
 - بازیابی شاخص های کنترلی، نظارت بر جریان و تسهیم اطلاعات در گردش کار سیستم
 - مقایسه عملکردهای واقعی و موارد برنامه ریزی شده
 - ارزیابی عملکردها بر اساس شاخص های تعیین شده
 - پایش و کنترل سازگاری بودجه های درآمدی، هزینه ای و نقدی با یکدیگر و با چارچوب های زمانی پروژه
 - تعیین انحراف ها و تصمیم گیری درمورد ضرورت ارایه پیشنهادات تغییرات مجدد
 - تاثیرگذاری بر (اصلاح) عوامل بروز تغییر
 - کنترل یکپارچه تغییرات و تایید تغییرات موردنیاز
 - پایش و ارزیابی نتایج اعمال تغییرات تایید شده
 - تعامل و اطلاع رسانی ذینفعان درمورد تغییرات و ارزیابی ها

- پیش بینی آینده پروژه سیستم و پایش ها و کنترل های موردنیاز محتمل
- ارزیابی کیفی قابلیت تولید، قابلیت استفاده، قابلیت اطمینان، قابلیت نگه داشت، قابلیت دسترسی، قابلیت کاربری، قابلیت انعطاف ...
- اولویت گذاری رویکرد های پیش گیرانه نسبت به روش های اصلاحی
- بررسی تعادل و توازن اهداف کیفی، زمانی و هزینه ای با خط مشی پروژه و اهداف
- تدقیق جزئیات ساختار شکست کار پروژه

چک لیست ارزیابی کیفی

ارزیابی کیفیت نرم افزار پروژه نظام مندی است که تضمین می نماید نرم افزار توسعه یافته از ابعاد کیفیتی مطلوبی برخوردار است. مدیریت کیفیت مجموعه ای از فعالیت ها و فرآیندهای منظم برنامه ریزی شده برای ایجاد و کنترل و اطمینان از کیفیت است [۲۲]. کیفیت نرم افزار دارای جنبه های مختلفی از قبیل قابلیت اطمینان، پایداری، ایمنی و غیره است. مدیریت موثر کیفیت موجب افزایش بهره وری تیم توسعه، بهبود کیفیت محصول، کاهش هزینه های کار مجدد به دلیل شناسایی زود هنگام نقص ها، افزایش سطح اطمینان و افزایش اعتبار نرم افزار و جلب اعتماد و رضایت مشتری می گردد. به منظور نیل به این اهداف فوق لازم است ارزیابی کیفی نرم افزار در تمامی مراحل توسعه نرم افزار رعایت گردد [۲۳-۲۴]. چک لیست ارزیابی کیفی به عنوان چک لیست مهم و اساسی دیگر در پکیج SDWCP و با هدف کنترل ابعاد کیفی نرم افزار تحت توسعه مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود:

چک لیست کنترل یکپارچگی

یکپارچگی در مهندسی نرم افزار به معنای ترکیب قطعات نرم افزاری و زیر سیستم ها در یک سیستم نرم افزاری جامع و ساخت یافته است. سیستم های یکپارچه، عملکرد بهتری را در مقایسه با مجموع تمامی سیستم های مستقل ارائه می دهند [۲۵]. آنچه اهمیت دارد، این است که یک برنامه کاربردی جامع، محتوای کاربردی بیشتری را ارائه می دهد. در این حالت، توزیع و استفاده از قابلیت های نرم افزار نیز آسان تر می شود. مدیریت یکپارچه سازی، به معنی ایجاد هماهنگی میان تمامی عناصر یک پروژه، از جمله وظایف، منابع، ذینفعان و آیتم های قابل تحویل است. این روند، راهکاری برای ایجاد هماهنگی و کارکرد فرآیندهای مختلف با یکدیگر ارائه می دهد. در واقع، مدیریت یکپارچگی فرآیندهای متعددی را که در یک پروژه مورد استفاده قرار می گیرند، بررسی کرده و از هماهنگی میان آنها مطمئن می شود. همچنین زمانی که پروژه های شما دارای اهداف رقابتی یا تضادهای زمان بندی هستند، مدیریت یکپارچه سازی کمک می کند تا تخصیص هزینه ها و زمان بندی فعالیت ها را به صورتی انجام شود که تیم توسعه قادر باشد، علاوه بر تکمیل پروژه، انتظارات ذینفعان را نیز برآورده کند [۲۶-۲۷]. چک لیست کنترل یکپارچگی به عنوان چک لیست دیگر در پکیج SDWCP و با هدف ممیزی یکپارچگی کارکردها، مشخصه های اجرائی و عملکردی در نرم افزار تحت توسعه مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود:

- تبیین مجدد منشور پروژه و ممیزی و تدقیق و تنظیم اهداف، نیازمندی ها و کارکردها با دستاوردهای سیستم
- انتخاب طرح یکپارچه سازی سیستم (مدل یکپارچگی ستاره، افقی، عمودی و اجرای فرمت مشترک داده)
- بررسی هماهنگی قابلیت های عملیاتی و اجرائی انجام شده با الزامات و نیازمندی ها و مدیریت پیکربندی سیستم
- بازیابی و کنترل بیانیه ارزیابی و تنظیم گستره گردش کار در سیستم و زیرسیستم ها
- مرور خط مبناهای پروژه برای اطمینان یکپارچگی
- عقب گرد و تکرار برخی فرآیندها در صورت ضرورت
- تبیین و تعیین استاندارد ها، روال ها، فرآیندها و شاخص های ارزیابی کیفی
- آدرس دهی مشخصه های کیفی لازم به الزامات مرتبط با محصول و پروژه
- تعریف محک و معیارهای پذیرش برای هر یک از اقلام تحویلی پروژه (شامل الزامات عملکردی و شرایط ویژه و ضروری)
- تهیه برنامه بهبود کیفی روال ها و فرآیندها
- کنترل و تفکیک مسئولیت های تضمین کیفیت پروژه بین اعضا و تیم های کاری درگیر پروژه
- الزامات مشتری / کارفرما باید در ارزیابی و تحلیل کیفی مورد توجه قرار گیرد و در تنظیم برنامه کیفیت به خواسته های مشتری / کارفرما به طور ویژه آدرس دهی شود
- تدوین ابزارهای سنجش و استانداردهای اختصاصی پروژه در صورت ضرورت و تدوین روش تحقق استاندارد ها
- تعیین شیوه ارزیابی ها (انطباق / عدم انطباق معیارها با استاندارد ها)
- ممیزی کیفیت و تفکیک و سنجش معیارها با فرضیات و اهداف از پیش تعیین شده
- بررسی نرم افزار و اسناد مرتبط از نظر عدم انطباق با استانداردها

هم پروژه های نرم افزاری تحت توسعه قابل بکارگیری می باشد. چک لیست پیشنهادی ابزاری درجهت ایجاد روال مدیریتی منظم جهت ارزیابی انجام وظائف و گردش کار ساخت یافته در توسعه سیستم های نرم افزاری می باشد. با بهره گیری از این پکیج می توان موثر بودن سیستم را به نحو قابل توجهی افزایش داد و از خطاهای احتمالی اجرائی و مدیریتی پرهیز نمود.

نتیجه گیری

توسعه نرم افزار فعالیتی است که از انواع پیشرفت های فناورانه استفاده کرده و نیازمند داشتن سطح بالایی از دانش است. از آنجائی که هر پروژه توسعه نرم افزار شامل تعداد متعددی از پارامترهای غیرقطعی، لازم است کنترل و نظارت بر گردش کار و اقدامات اجرائی در فرایند توسعه سیستم بصورت مدون و تفضیلی صورت پذیرد. چک لیست ها ابزار مدیریتی کمکی درجهت تدقیق جریان امور و کنترل عملیات ایجاد می نمایند و امروز در کاربردهای متنوع مورد بهره برداری قرار می گیرند. به دلیل اهمیت وجود یک فرایند جامع و ساخت یافته در توسعه یک سیستم نرم افزاری و در این راستا، در این مقاله چک لیست مدیریتی به نام چک لیست گردش کار توسعه سیستم نرم افزاری را پیشنهاد و اجزاء آن را توصیف نمودیم. این چک لیست درواقع یک پکیج حاوی زیرمجموعه ای از چک لیست های جزئی تر نظارتی در فرایند توسعه سیستم نرم افزاری می باشد. چک لیست پیشنهادی ابزاری درجهت ایجاد روال مدیریتی منظم جهت ارزیابی انجام وظائف و گردش کار ساخت یافته در توسعه سیستم های نرم افزاری می باشد. چک لیست گردش کار پیشنهادی علاوه بر تبیین گردش کار سیستم و شفاف سازی نظارت بر توسعه سیستم نرم افزاری، به عنوان منبعی برای اعضای تیم های کاری در جهت حل و رفع اختلاف نظرات و همسوسازی اقدامات اجرائی مطرح می شود و همچنین برای پرهیز و جلوگیری از اشتباهات رایج مدیریتی قابل بکارگیری می باشد.

منابع

- [1] Molléri JS, Petersen K, Mendes E. Survey guidelines in software engineering: An annotated review. In Proceedings of the 10th ACM/IEEE international symposium on empirical software engineering and measurement 2016 Sep 8 (pp. 1-6).
- [2] Zhang L, Tian JH, Jiang J, Liu YJ, Pu MY, Yue T. Empirical research in software engineering—a literature survey. Journal of Computer Science and Technology. 2018 Sep;33:876-99.
- [3] Bhuvanewari T, Prabakaran S. A survey on software development life cycle models. International Journal of Computer Science and Mobile Computing. 2013 May;2(5):262-7.

- متعادل سازی طرح یکپارچه سازی سیستم
- انجام فرایند تایید رسمی محصول
- تکمیل گزارش پیشرفت یکپارچگی و تایید عملکرد نهایی
- آیتم های قابل تحویل سیستم

چک لیست مستندات

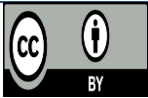
مستندسازی نرم افزار، شامل مجموعه ای از مستندات مکتوب است که همراه نرم افزارها ارائه می گردد. در صنعتی مانند توسعه نرم افزار، مستند سازی باید کاملاً جامع باشد، زیرا هرگونه خطایی سبب بروز اختلاف میان افراد خواهد شد. مستند سازی، اطلاعات مربوط به هر فاز چرخه حیات پروژه را به طور واضح ترسیم می کند و از سوءاستفاده و سوءتفاهم ها جلوگیری می کند. قسمت اعظمی از هزینه پروژه های نرم افزاری را مستندسازی نرم افزار تشکیل داده و به همین دلیل از اهمیت خاصی در تولید برنامه های نرم افزاری برخوردار است [۲۸]. شرکت های نرم افزاری جهت ایجاد و بروز رسانی مستندات کلیه محصولات خود ردیف شغلی کارشناس مستندسازی و استقرار نرم افزار ایجاد نموده اند تا با تهیه مستندات و محتوای آموزشی مطلوب در هنگام نیاز ساختار و قالب مناسبی از سامانه های نرم افزاری خود داشته باشند. اجرای درست و مطلوب پروژه های نرم افزاری به کیفیت و کامل بودن مستندات پیش تولید بستگی دارد زیرا در توسعه برنامه های نرم افزاری همواره نیاز است تا کارشناس توسعه نرم افزار به شکل متناوب، به مستندات آنچه سیستم باید انجام دهد، مراجعه نماید. یکی از شیوه های تولید مستندات و جمع آوری اطلاعات، تکمیل مستندات به طور تدریجی و مرحله ای می باشد، هرچند این روش خالی از ایراد نیست [۲۹]. چک لیست مستندسازی به عنوان چک لیست دیگر در پکیج SDWCP و با هدف کنترل مستندسازی سیستم و نرم افزار تحت توسعه مورد بهره برداری قرار می گیرد. این چک لیست با در نظر گرفتن نکات و ملاحظات زیر انجام می شود و می توان از موارد نام برده شده به عنوان زیرمجموعه دیگری در پکیج SDWCP استفاده نمود:

- بررسی و ارزیابی مستند تعریف پروژه، توصیف محصول
- بررسی و ارزیابی مستند نیازمندی ها و مدیریت پیکربندی
- تدقیق مستندات الگوها و طراحی ها
- بررسی و ارزیابی مستند طراحی معماری
- بررسی و ارزیابی مستندات مشخصه های فنی و عملیاتی سیستم و جریان های کاری
- بررسی و ارزیابی مستندات آزمون و خطایابی
- تطابق مستندات بودجه، هزینه و چارچوب های زمانی و اجرائی پروژه
- گزارش استانداردها در مستند مدیریت کیفیت
- آرشینو نتایج بررسی ها و گزارش دهی

در پایان لازم به ذکر است، بهره گیری از پکیج چک لیست گردش کار پیشنهادی در فرایند توسعه سیستم هم برای پروژه های در حال اجرا و

- [15] Madaio MA, Stark L, Wortman Vaughan J, Wallach H. Co-designing checklists to understand organizational challenges and opportunities around fairness in AI. In Proceedings of the 2020 CHI conference on human factors in computing systems 2020 Apr 21 (pp. 1-14).
- [16] Lewis WE. Software testing and continuous quality improvement. CRC press; 2017 Jun 1.
- [17] Kupiainen E, Mäntylä MV, Itkonen J. Using metrics in Agile and Lean Software Development—A systematic literature review of industrial studies. Information and software technology. 2015 Jun 1;62:143-63.
- [18] Wieringa RJ. Design science methodology for information systems and software engineering. Springer; 2014 Nov 19.
- [19] Wagner S, Ruhe M. A systematic review of productivity factors in software development. arXiv preprint arXiv:1801.06475. 2018 Jan 19.
- [20] Blank S, Dorf B. The startup owner's manual: The step-by-step guide for building a great company. John Wiley & Sons; 2020 Mar 17.
- [21] Braude EJ, Bernstein ME. Software engineering: modern approaches. Waveland Press; 2016 Mar 9.
- [22] Dennis A, Wixom B, Tegarden D. Systems analysis and design: An object-oriented approach with UML. John wiley & sons; 2015 Mar 2.
- [23] Ghobadi S. What drives knowledge sharing in software development teams: A literature review and classification framework. Information & Management. 2015 Jan 1;52(1):82-97.
- [24] Mayring P. Qualitative content analysis: theoretical foundation, basic procedures and software solution.
- [25] Ilyas M, Khan SU, Rashid N. Empirical validation of software integration practices in global software development. SN Computer Science. 2020 May;1:1-23.
- [26] Lasso A, Kazanzides P. System integration. In Handbook of Medical Image Computing and Computer Assisted Intervention 2020 Jan 1 (pp. 861-891). Academic Press.
- [4] Sarker IH, Faruque F, Hossen U, Rahman A. A survey of software development process models in software engineering. International Journal of Software Engineering and Its Applications. 2015 Nov;9(11):55-70.
- [5] Suleiman H, Adepetu A, Arnautovic E, Svetinovic D. Comprehensive integrated checklists for requirements engineering and software project management. In 2013 International Conference on Information Science and Applications (ICISA) 2013 Jun 24 (pp. 1-4). IEEE.
- [6] Brace W, Ekman K. CORAMOD: a checklist-oriented model-based requirements analysis approach. Requirements Engineering. 2014 Mar;19:1-26.
- [7] Morschheuser B, Hassan L, Werder K, Hamari J. How to design gamification? A method for engineering gamified software. Information and Software Technology. 2018 Mar 1;95:219-37.
- [8] Kanewala U, Bieman JM. Testing scientific software: A systematic literature review. Information and software technology. 2014 Oct 1;56(10):1219-32.
- [9] Houston D. The value of a good checklist. Software Quality Professional. 2004 Mar 1;6(2):17.
- [10] Tausworthe RC. The work breakdown structure in software project management. Journal of Systems and Software. 1979 Jan 1;1:181-6.
- [11] Li L. The impact of risk checklists on project manager's risk perception and decision-making process.
- [12] Marcelino-Sádaba S, Pérez-Ezcurdia A, Lazcano AM, Villanueva P. Project risk management methodology for small firms. International journal of project management. 2014 Feb 1;32(2):327-40.
- [13] Parmenter D. Key performance indicators: developing, implementing, and using winning KPIs. John Wiley & Sons; 2015 Apr 13.
- [14] Kuhrmann M, Diebold P, Münch J, Tell P, Garousi V, Felderer M, Trektene K, McCaffery F, Linsen O, Hanser E, Prause CR. Hybrid software and system development in practice: waterfall, scrum, and beyond. In Proceedings of the 2017 international conference on software and system process 2017 Jul 5 (pp. 30-39).

- on Software Engineering (ICSE) 2019 May 25 (pp. 1199-1210). IEEE.
- [29] Al-Saqqa S, Sawalha S, AbdelNabi H. Agile software development: Methodologies and trends. International Journal of Interactive Mobile Technologies. 2020 Nov 1;14(11).
- [27] Kotha S, Gopal TV. Formal methods for enterprise application integration. Complex Control System. 2021;3(1):9-24.
- [28] Aghajani E, Nagy C, Vega-Márquez OL, Linares-Vásquez M, Moreno L, Bavota G, Lanza M. Software documentation issues unveiled. In2019 IEEE/ACM 41st International Conference



COPYRIGHTS

©2021 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, as long as the original authors and source are cited. No permission is required from the authors or the publishers.