



فهرست مسائل

۱	مسئله ۱
۱	بخش اول
۳	بخش دوم
۶	مسئله ۲
۶	بخش اول
۷	بخش دوم
۷	بخش سوم
۸	بخش چهارم
۹	بخش پنجم
۹	بخش ششم
۱۱	مسئله ۳
۱۱	بخش اول
۱۱	بخش دوم
۱۲	بخش سوم
۱۳	بخش چهارم
۱۳	بخش پنجم

پاسخ مسئله‌ی ۱.

بخش اول

با توجه به اسلایدها، سلسله اقدامات پیشنهادی از آغاز تا تحویل محصول را به صورت گام به گام شرح می‌دهم:

۱. ارتباطات

- اقدام اولیه: اولین گام درک عمیق نیازها و خواسته‌های شرکت است. برای این منظور، جلسه‌ای با ذینفعان اصلی پروژه برگزار می‌شود تا دیدی واضح از چشم‌انداز پروژه و اهداف اصلی آنها در پیاده‌سازی این فناوری به دست آوریم.
- تعریف ذینفعان: شناسایی افرادی که در پروژه نقشی دارند و بررسی دقیق نیازها، محدودیت‌ها و انتظارات آنان، که شامل مدیران، تیم فنی و بازاریابی شرکت فرش گستران سنت می‌شود.
- گردآوری اطلاعات نامشخص: در این مرحله، جزئیاتی مانند مشخصات دقیق تکنولوژی‌های مورد نظر، انتظارات از محصول نهایی، و محدودیت‌های فنی و بودجه‌ای بررسی می‌شود.
- نمایش گرافیکی مشکل: اگر امکان‌پذیر باشد، می‌توان از نمودارها و مدل‌های گرافیکی برای نمایش بهتر نیازمندی‌ها و الزامات اولیه استفاده کرد.

۲. مدل سازی و طراحی اولیه

- طرح یک راه حل: با استفاده از اطلاعات به دست آمده، قدم بعدی تدوین یک برنامه ریزی دقیق است. در این مرحله طراحی مدل اولیه صورت می گیرد که شامل تعیین معماری نرم افزار، ابزارهای هوش مصنوعی مورد استفاده، و سایر فناوری ها است.
- بررسی موارد مشابه: تجزیه و تحلیل نمونه های موفق مشابه، مانند استفاده از هوش مصنوعی در سایر صنایع، به تعیین الگوها و روش های مفید کمک می کند.
- بررسی راه حل های موجود: بررسی ابزارها و نرم افزارهای موجود که می توانند در این پروژه مورد استفاده قرار گیرند و در صورت نیاز به سفارشی سازی آنها.
- تعریف زیرمسئله ها: به منظور تسهیل توسعه، مسئله اصلی به چندین زیرمسئله تقسیم می شود که هر یک به طور جداگانه و به صورت ماژولار قابل حل باشد.

۳. کدنویسی و توسعه محصول

- پیاده سازی: پس از تایید طرح کلی، تیم توسعه کار بر روی ساخت و کدنویسی سیستم را آغاز می کند.
- تطابق کد با طراحی اولیه: تیم توسعه، طراحی و معماری تعریف شده را به کد تبدیل می کند و از این طریق اطمینان حاصل می شود که کد با مدل های ارائه شده در مرحله قبلی مطابقت دارد.
- بررسی و صحت سنجی هر بخش: با استفاده از بررسی های داخلی و استفاده از کدهای تست اولیه، تیم تلاش می کند تا هر مرحله از کدنویسی را مورد آزمایش قرار دهد. در صورت وجود خطا، تیم آنها را اصلاح و مجدداً بررسی می کند.
- آزمایش ها و تصحیح ها: از روش هایی مانند تست واحد برای ارزیابی اجزای کوچک و تست های یکپارچگی برای ارزیابی ارتباط بین ماژول ها استفاده می شود تا اطمینان حاصل شود که سیستم به درستی کار می کند و همه عملکردهای اساسی به طور دقیق اجرا می شوند.

۴. تست و کنترل کیفیت

- آزمایش و اطمینان از کیفیت: بررسی نهایی با هدف اطمینان از دقت و کارایی نرم افزار انجام می شود و معیارهای کیفیت به دقت کنترل می شود.
- تست عملکرد سیستم: انجام تست های نهایی برای اطمینان از اینکه سیستم به تمام نیازهای تعریف شده در اسناد اولیه پاسخ می دهد.
- ارزیابی کیفیت فنی و امنیتی: بررسی مواردی چون امنیت، کارایی و استحکام سیستم برای اطمینان از اینکه محصول نهایی برای استفاده در محیط واقعی آماده است.
- برگزاری جلسه بازنگری با ذینفعان: نمایش محصول اولیه و دریافت بازخورد از طرف ذینفعان برای رفع مشکلات و برطرف کردن نیازهای احتمالی.

۵. تحویل و استقرار محصول

- انتقال و استقرار محصول: پس از انجام تمام تست ها و اطمینان از آمادگی کامل سیستم، محصول نهایی به شرکت تحویل داده می شود.
- آموزش و پشتیبانی کاربران: آموزش تیم فنی شرکت و ارائه اسناد و دستورالعمل های لازم برای استفاده از سیستم.

- استقرار در محیط عملیاتی: نصب و راه اندازی سیستم در محیط واقعی، اطمینان از عملکرد صحیح آن و حل مشکلات احتمالی که ممکن است در این مرحله بروز کنند.
- پشتیبانی پس از استقرار: برای پشتیبانی از کاربران و رفع مشکلات احتمالی، تیم پشتیبانی باید به مدت معین پس از تحویل محصول در دسترس باشد و پشتیبانی فنی ارائه دهد.

بخش دوم

برای رسیدن به یک برنامه مناسب، ابتدا باید به این پرسش ها پاسخ دهیم:

- چگونه می توان از عقب ماندگی کاست و رقابت را از دست نداد؟
- آیا می توان از قابلیت های فعلی پروژه بهره بیشتری برد؟

مسیر فکری و اقدامات پیشنهادی به شرح زیر هستند، این رویکرد نه تنها باعث تسریع در توسعه و عرضه محصول می شود، بلکه از دیدگاه مشتریان، محصولی نوآورانه و به روز به بازار عرضه می کند که توانایی رقابت با محصول رقیب را دارد:

۱. ارزیابی وضعیت کنونی پروژه

مسیر فکری:

- ابتدا باید بدانیم پروژه تا چه اندازه به اهداف تعریف شده نزدیک است و چه میزان از آن تکمیل شده است. این امر به شناسایی نقاط ضعف و قوت ما در مقایسه با محصول رقیب کمک می کند.

اقدامات:

- بررسی جزئیات پروژه در شش ماه گذشته: تهیه گزارشی از مراحل انجام شده، نقاط قوت و ضعف، موانع و دستاوردها.
- تحلیل قابلیت های محصول در مقایسه با محصول دیگر: در این مرحله باید ببینیم آیا ویژگی هایی که تاکنون طراحی کرده ایم، رقابتی هستند یا خیر. این تحلیل به ما کمک می کند که تصمیم بگیریم آیا به توسعه فعلی ادامه دهیم یا تغییراتی در ویژگی های کلیدی محصول اعمال کنیم.

۲. بهبود و بازنگری در قابلیت ها

مسیر فکری:

- با توجه به محوریت هوش مصنوعی در محصول Airpet، یکی از اولویت های مهم این است که قابلیت های هوش مصنوعی پروژه را تقویت کنیم و امکانات نوآورانه تری نسبت به رقیب ارائه دهیم.

اقدامات:

- ایجاد تیم تخصصی هوش مصنوعی: تشکیل تیمی کوچک و متخصص در حوزه هوش مصنوعی برای تمرکز بر ویژگی های اختصاصی و پیشرفته تر.
- همکاری با متخصصان و مشاوران AI خارجی یا داخلی: استفاده از مشاوران خارجی یا متخصصانی که پیشتر در پروژه های مشابه کار کرده اند، می تواند به شتاب دهی پروژه کمک کند و نوآوری های جدیدی را به محصول اضافه کند.

- افزودن قابلیت‌های هوش مصنوعی پیشرفته‌تر و شخصی‌سازی شده: بررسی و پیاده‌سازی قابلیت‌هایی نظیر تشخیص الگوهای فروش توسط هوش مصنوعی، تنظیمات تطبیق‌پذیر بر اساس سلیقه مشتری، و قابلیت پیشنهاد بر اساس بازخورد کاربر.

۳. بازنگری و تسریع در فرآیند توسعه

مسیر فکری:

- با توجه به محدودیت زمانی و عقب‌ماندگی از رقیب، باید فرآیند توسعه را تسریع کرد. در این مسیر می‌توان از برخی روش‌های چابک و نوآورانه استفاده کرد.

اقدامات:

- استفاده از روش‌های توسعه چابک (Agile): برای افزایش سرعت توسعه، پیشنهاد می‌شود از روش‌های چابک مانند اسکرام (Scrum) بهره ببریم. تقسیم پروژه به «اسپرینت‌ها» و اهداف کوتاه‌مدت به جای تمرکز بر روی اهداف بلندمدت کمک می‌کند تا سریع‌تر به دستاوردهای قابل تحویل دست یابیم.
- تقسیم تیم‌ها و توزیع وظایف به صورت هم‌زمان: تجزیه و تقسیم پروژه به تیم‌های کوچک‌تر و تخصصی‌تر که بتوانند به صورت موازی کار کنند و برخی از وظایف را به طور هم‌زمان انجام دهند.

۴. استفاده از قابلیت MVP

مسیر فکری:

- با توجه به محدودیت زمانی، می‌توان ابتدا یک محصول اولیه با قابلیت‌های اصلی به بازار عرضه کرد و پس از دریافت بازخورد، به توسعه و بهبود آن پرداخت.

اقدامات:

- تعیین MVP با ویژگی‌های کلیدی: شناسایی مهم‌ترین و پرکاربردترین قابلیت‌های محصول و تمرکز روی توسعه و عرضه این قابلیت‌ها به صورت اولیه.
- آزمایش اولیه MVP و دریافت بازخورد کاربران: از طریق آزمایش و بازخورد سریع از کاربران، می‌توان محصول را بهبود بخشید و تغییرات لازم را سریع‌تر اعمال کرد.

۵. کمپین بازاریابی و اطلاع‌رسانی رقابتی

مسیر فکری:

- با توجه به رونمایی محصول Airpet، باید اطلاع‌رسانی مناسبی از برنامه‌ها و ویژگی‌های منحصر به فرد محصول «مهر فرش گستران سنت» صورت گیرد تا مشتریان به این باور برسند که محصول جدید همچنان می‌تواند پاسخگوی نیازهای آنان باشد.

اقدامات:

- راه‌اندازی کمپین اطلاع‌رسانی در مورد ویژگی‌های متمایز: اطلاع‌رسانی دقیق و تبلیغاتی که به وضوح ویژگی‌های متمایز محصول و مزایای رقابتی آن را در مقابل محصول رقیب به نمایش بگذارد.

- استفاده از بازخوردهای اولیه کاربران برای تبلیغ محصول: با جمع آوری و نمایش بازخوردهای مثبت کاربران اولیه محصول، می توان اعتماد بیشتری از مشتریان جدید کسب کرد.
- ایجاد یک طرح پیش فروش برای جلب توجه بازار: این اقدام می تواند باعث افزایش هیجان و استقبال بازار شود و مشتریان را به انتظار برای محصول نهایی ترغیب کند.

۶. برنامه ریزی برای انتشار مداوم بهبودها و قابلیت های جدید

مسیر فکری:

- پس از عرضه اولیه محصول، باید به طور مداوم قابلیت ها و بهبودهای جدیدی به محصول اضافه شود تا جذابیت و کاربردپذیری محصول در طول زمان افزایش یابد.

اقدامات:

- زمان بندی به روزرسانی های دوره ای: برای حفظ رقابت پذیری، برنامه ریزی منظمی برای انتشار به روزرسانی ها و قابلیت های جدید در بازه های زمانی مشخص صورت گیرد.
- جذب بازخورد کاربران و استفاده از آن در برنامه های توسعه بعدی: تحلیل نظرات و بازخوردهای کاربران در جهت افزودن قابلیت های کاربردی تر و حذف یا بهبود موارد ناکارآمد.

پاسخ مسئله‌ی ۲.

بخش اول

شباهت‌ها

۱. توجه مستمر به ارزیابی و مدیریت ریسک در هر تکرار:

هر دو مدل مارپیچی و فرایند یکپارچه به صورت تکرارشونده طراحی شده‌اند و در هر چرخه، ارزیابی ریسک صورت می‌گیرد. در مدل مارپیچی، هر دور از مارپیچ با ارزیابی ریسک همراه است، که این امر به اتخاذ تصمیم‌های مناسب قبل از حرکت به مرحله بعد کمک می‌کند. در فرایند یکپارچه، هر تکرار شامل ارزیابی مشکلات و ریسک‌ها است که امکان کنترل و بهبود مستمر را فراهم می‌کند.

۲. ساختار انعطاف‌پذیر و واکنش‌پذیر نسبت به تغییرات و رفع مشکلات:

هر دو مدل با طراحی تکرارشونده امکان تطبیق با تغییرات و حل مشکلات پیش‌بینی نشده را فراهم می‌کنند. در پروژه‌هایی با نیازهای پویا و حساسیت بالا مانند سیستم پزشکی، این ویژگی باعث می‌شود که تیم بتواند با اطمینان بیشتری ریسک‌ها را مدیریت کند و در هر تکرار بازخورد لازم برای کاهش ریسک‌ها را به دست آورد.

تفاوت‌ها

۱. نحوه ارزیابی ریسک در طول فرآیند توسعه:

- مدل مارپیچی: این مدل بر ارزیابی مداوم و پیوسته ریسک‌ها در هر دور از چرخه توسعه تمرکز دارد. در هر تکرار، ارزیابی ریسک‌ها به عنوان بخش اصلی فرآیند مارپیچ انجام می‌شود و هرگونه تغییر در اولویت‌ها و تصمیم‌گیری‌ها مبتنی بر شناسایی ریسک‌های جدید است. این روش به تیم اجازه می‌دهد که در هر مرحله ریسک‌های به‌روز و مرتبط با هر فاز را شناسایی کرده و استراتژی مناسبی برای کاهش آنها انتخاب کند.
- فرایند یکپارچه: در این مدل، ارزیابی ریسک در مراحل اولیه پروژه مانند فاز "شروع" و "تکامل" انجام می‌شود، جایی که ریسک‌های اصلی و تاثیرگذار شناسایی شده و معماری و برنامه‌ریزی کلی با در نظر گرفتن این ریسک‌ها پی‌ریزی می‌شود. برخلاف مدل مارپیچی، در فرایند یکپارچه پس از مراحل اولیه، مدیریت ریسک به‌طور مستقیم در هر تکرار دنبال نمی‌شود و بیشتر به عنوان بخشی از فعالیت‌های پشتیبان در چارچوب کلی پروژه در نظر گرفته می‌شود.

۲. نقش مدیریت ریسک در تصمیم‌گیری‌ها و جهت‌دهی به پروژه:

- مدل مارپیچی: مدیریت ریسک در این مدل به عنوان عنصر مرکزی و راهبردی در هر تکرار عمل می‌کند و تصمیمات کلیدی برای ادامه یا تغییر مسیر در پایان هر چرخه بر اساس ارزیابی ریسک‌های شناسایی شده اتخاذ می‌شود. این امر موجب می‌شود که پروژه به‌طور پویا به تغییرات پاسخ دهد و تیم بتواند در هر مرحله بر اساس تحلیل دقیق ریسک‌ها اقدامات اصلاحی را انجام دهد.
- فرایند یکپارچه: در این مدل، مدیریت ریسک در تصمیم‌گیری‌ها و برنامه‌ریزی‌های اولیه اهمیت دارد، اما پس از تعیین چارچوب معماری و ساختار پروژه در مراحل اولیه، نقش مستقیم آن کاهش می‌یابد. تصمیمات پروژه عمدتاً بر اساس طراحی و معماری تثبیت شده اتخاذ می‌شود و مدیریت ریسک به عنوان یک فعالیت پشتیبان در بهبود مستمر کیفیت و تغییرات جزئی محصول در طول چرخه زندگی پروژه باقی می‌ماند.

بخش دوم

مدل مارپیچی

- شناسایی ریسک:

هر دور از چرخه مارپیچی با ارزیابی و شناسایی ریسک‌ها آغاز می‌شود. این شناسایی شامل بررسی ریسک‌های فنی، زمانی، مالی، و عملکردی است که ممکن است بر روند توسعه تأثیر بگذارد. تیم توسعه به طور مستمر در هر تکرار به دنبال شناسایی ریسک‌های جدید و بررسی عوامل بالقوه‌ای است که ممکن است تهدیدی برای موفقیت پروژه باشند.

- تحلیل ریسک:

پس از شناسایی ریسک‌ها، مدل مارپیچی بر تحلیل عمیق این ریسک‌ها تمرکز دارد. تیم توسعه در هر تکرار میزان تأثیر و احتمال وقوع هر ریسک را ارزیابی می‌کند. این تحلیل به تیم کمک می‌کند که ریسک‌ها را اولویت‌بندی کند و منابع بیشتری را به ریسک‌های مهم‌تر و محتمل‌تر اختصاص دهد.

- کاهش ریسک:

در این مرحله، بر اساس نتایج تحلیل، تیم به دنبال طراحی راه‌حلی برای کاهش یا رفع ریسک‌های شناسایی شده است. اقدامات کاهش‌دهنده می‌تواند شامل آزمایشات اولیه، ساخت نمونه‌های اولیه و استفاده از تکنیک‌های خاص برای بررسی صحت عملکرد باشد. در هر تکرار، این مدل به تیم اجازه می‌دهد که هر مرحله را با تمرکز بر کاهش ریسک تکمیل کند، پیش از آنکه به مراحل بعدی برود. این فرآیند به خصوص برای پروژه‌هایی با ریسک‌های زیاد و متغیر، مانند سیستم‌های حساس به داده‌های پزشکی، مفید است.

مدل فرایند یکپارچه

- شناسایی ریسک:

در فازهای اولیه مانند فاز شروع و تکامل، فرایند یکپارچه به شناسایی ریسک‌های کلی و اساسی پروژه می‌پردازد. این شناسایی شامل ریسک‌های مرتبط با نیازهای مشتری، محدودیت‌های فنی و مشکلات احتمالی در پیاده‌سازی می‌شود. در این مراحل، تمرکز اصلی بر شناسایی ریسک‌هایی است که می‌توانند در بلندمدت بر کل پروژه تأثیر بگذارند.

- تحلیل ریسک:

فرایند یکپارچه به تحلیل ریسک‌ها در فازهای آغازین پرداخته و با استفاده از این تحلیل، چارچوب و معماری پروژه را طراحی می‌کند. این تحلیل به تیم توسعه کمک می‌کند تا نیازها و محدودیت‌های اصلی پروژه را به درستی ارزیابی کرده و معماری سیستم را به گونه‌ای طراحی کند که ریسک‌های کلیدی را از ابتدا کنترل کند.

- کاهش ریسک:

پس از تحلیل، فرایند یکپارچه از طریق ایجاد یک معماری قابل اعتماد و پایدار در مراحل اولیه به کاهش ریسک‌ها می‌پردازد. علاوه بر این، در هر تکرار به بررسی مجدد و بهینه‌سازی معماری و طراحی سیستم پرداخته می‌شود. اما برخلاف مدل مارپیچی، فرایند یکپارچه به طور مداوم بر شناسایی و کاهش ریسک‌ها در طول هر تکرار تمرکز ندارد، بلکه تمرکز اصلی آن بر طراحی صحیح و جامع اولیه است که بتواند بیشتر ریسک‌ها را در درازمدت کاهش دهد.

بخش سوم

با توجه به نیاز به تطابق با استانداردهای سخت‌گیرانه، که محافظت از حریم خصوصی و امنیت داده‌های بیماران را الزامی می‌کند، مدل مارپیچی گزینه مناسب‌تری برای این پروژه خواهد بود. زیرا:

مدیریت دقیق و پیوسته ریسک

مدل مارپیچی بر اساس یک چرخه تکرارشونده طراحی شده است که در هر دور آن، ارزیابی و کاهش ریسک‌های جدید انجام می‌شود. برای پروژه‌ای که شامل داده‌های حساس پزشکی و نیازمند حفظ حریم خصوصی بیماران است، این ویژگی بسیار مهم است. این مدل به تیم توسعه امکان می‌دهد تا در هر تکرار:

- ریسک‌های جدید را شناسایی و تحلیل کند.
- ریسک‌های مربوط به امنیت و حفظ حریم خصوصی داده‌های بیماران را اولویت‌بندی و اقدامات لازم را برای کاهش آنها اتخاذ کند.

این بررسی‌های دوره‌ای، انعطاف لازم را برای انطباق با تغییرات احتمالی در استانداردها و قوانین امنیتی و حفظ حریم خصوصی فراهم می‌کند و باعث می‌شود تیم بتواند به سرعت به هر گونه تهدید یا ریسک جدید در طول چرخه توسعه واکنش نشان دهد.

انعطاف‌پذیری در پیاده‌سازی امنیت و حریم خصوصی

مدل مارپیچی به دلیل تمرکز بر کاهش ریسک در هر مرحله، به تیم اجازه می‌دهد که برای موارد خاص مانند تطابق با HIPAA یا کنترل‌های امنیتی خاص، اقدامات خاصی را در هر تکرار به کار گیرد. این انعطاف‌پذیری به تیم کمک می‌کند تا در هر دور از چرخه، ویژگی‌های امنیتی و حفظ حریم خصوصی را بررسی کرده و در صورت نیاز آنها را بهبود بخشد. از آنجا که این پروژه نیاز به حفاظت از داده‌های حساس بیماران دارد، توجه مستمر به حریم خصوصی و امنیت در هر مرحله از توسعه می‌تواند از بروز آسیب‌پذیری‌های احتمالی جلوگیری کند.

بخش چهارم

در مدیریت تغییرات و استقرارهای مکرر، هر دو مدل افزایشی و چابک قابلیت‌ها و رویکردهای خاص خود را دارند:

مدل افزایشی

- مدیریت تغییرات: در مدل افزایشی، سیستم به صورت تدریجی و در چندین مرحله ساخته و توسعه داده می‌شود. هر بخش از سیستم به عنوان یک افزایش تعریف می‌شود که شامل ویژگی‌ها یا بخش‌هایی از عملکرد کلی پروژه است. در صورت بروز تغییرات جدید، می‌توان این تغییرات را در یکی از این "افزایش‌ها" برنامه‌ریزی کرد و سپس آن را در سیستم اصلی ادغام نمود. این مدل امکان مدیریت تغییرات را فراهم می‌کند، اما تغییرات جدید معمولاً پس از پایان هر افزایش و قبل از شروع افزایش بعدی در نظر گرفته می‌شود. بنابراین، مدیریت تغییرات در این مدل از انعطاف کمتری نسبت به مدل چابک برخوردار است.
- مدیریت استقرارهای مکرر: هر افزایش در مدل افزایشی به صورت مستقل توسعه و آزمایش می‌شود و پس از تکمیل، به سیستم اصلی افزوده می‌شود. استقرارهای مکرر در این مدل به این صورت انجام می‌شود که هر افزایش تکمیل شده به محصول نهایی اضافه و استقرار داده می‌شود. این روش برای پروژه‌هایی که نیاز به استقرارهای گام به گام دارند مناسب است، اما سرعت استقرارهای آن معمولاً کندتر از مدل چابک است زیرا باید هر افزایش به طور کامل تکمیل شود.

مدل افزایشی برای پروژه‌هایی که نیاز به استقرارهای مرحله‌ای و تغییرات کمتر دارند مناسب است. در این مدل تغییرات و به‌روزرسانی‌ها پس از تکمیل هر افزایش اعمال می‌شوند و استقرارهای مکرر به اندازه مدل چابک انعطاف‌پذیر نیست.

مدل چابک

- مدیریت تغییرات: در مدل چابک، تغییرات به عنوان بخشی از فرآیند توسعه پذیرفته و حتی تشویق می شوند. تیم ها از بازه های کوتاهی به نام اسپرینت استفاده می کنند که در هر اسپرینت، می توان تغییرات جدید را بررسی و در صورت نیاز، در برنامه ها اعمال کرد. این انعطاف پذیری بالا باعث می شود که تغییرات به طور پیوسته و بدون نیاز به بازنگری کل سیستم در نظر گرفته شوند و ویژگی های جدید به سرعت به محصول اضافه شوند.
- مدیریت استقرارهای مکرر: مدل چابک به دلیل استفاده از اسپرینت های کوتاه و توسعه مداوم، امکان استقرارهای مکرر و سریع را فراهم می کند. در پایان هر اسپرینت، یک نسخه عملیاتی و قابل استفاده از سیستم که شامل تغییرات و به روزرسانی های جدید است، ارائه می شود. این رویکرد استقرار مداوم به خصوص برای پروژه هایی با نیازهای متغیر و پویا مانند سیستم های پزشکی مناسب است که در آن ها به روزرسانی های سریع و رفع مشکلات بدون اختلال در عملکرد اصلی سیستم ضروری است.

مدل چابک به دلیل انعطاف بالا در مدیریت تغییرات و سرعت استقرارهای مکرر، برای پروژه های پویا که نیاز به تطبیق سریع با نیازهای جدید و به روزرسانی های امنیتی دارند، ایده آل است.

بخش پنجم

این بخش از مسئله در بخش واضح (Clear) چارچوب کانوین جای می گیرد. بخش واضح در چارچوب کانوین برای پروژه هایی مناسب است که راه حل های از پیش تعیین شده و قابل اطمینان دارند، نیاز به تحلیل پیچیده یا اکتشاف ندارند و از رویکردهای استاندارد و شناخته شده استفاده می کنند.

- شفافیت نیازمندی ها: در این پروژه، نیازمندی ها از پیش به طور کامل مشخص و شفاف هستند و سال ها است که ثابت دارند. این بدان معناست که هیچ ابهام یا تغییر غیرمنتظره ای در خواسته های سیستم وجود ندارد و تیم به طور دقیق می داند چه چیزی باید پیاده سازی شود.
- تجربه قبلی تیم: تیم فنی پروژه تجربه بالایی در زمینه پیاده سازی سیستم های مشابه دارد و پیش تر این نوع پروژه ها را انجام داده است. بنابراین، راه حل ها و فرآیندهای لازم برای اجرای پروژه به طور کامل آشنا هستند.
- دسترسی به روالهای مطلوب: در این پروژه، منابع کافی از روش های بهینه و روال های مطلوب برای پیاده سازی سیستم وجود دارد. این ویژگی مشخصاً نشان می دهد که بهترین راه حل ها از قبل در دسترس هستند و نیازی به کشف یا ابداع فرآیندهای جدید نیست.

بخش ششم

برای این پروژه خاص، مدل آبشاری به دلایل زیر بهترین گزینه است:

- نیازمندی های شفاف و پایدار: در مدل آبشاری، مراحل به صورت متوالی و خطی پیش می روند و هر مرحله پس از تکمیل مرحله قبلی آغاز می شود. این مدل برای پروژه هایی مناسب است که نیازمندی ها از ابتدا مشخص و پایدار هستند و به تغییرات زیادی نیاز ندارند. از آنجا که نیازمندی های این پروژه از سال ها پیش مشخص و شفاف بوده و احتمال تغییر در آنها بسیار کم است، مدل آبشاری به دلیل تمرکز بر طراحی و برنامه ریزی دقیق در مراحل اولیه، می تواند گزینه ای مؤثر باشد.

- تجربه قبلی تیم و وجود روال‌های مطلوب:
با توجه به اینکه تیم تجربه بالایی در اجرای پروژه‌های مشابه دارد و منابع کافی از روال‌های مطلوب برای این سیستم در دسترس است، می‌توان از مدل آشناری که رویکردی ساختارمند و استاندارد دارد استفاده کرد. این مدل با پیروی از روش‌های تعریف‌شده در مراحل مختلف، می‌تواند به تیم کمک کند تا با بهره‌گیری از دانش و تجربه قبلی به‌طور مؤثر به اهداف پروژه دست یابد.
- عدم نیاز به تکرار یا انعطاف بالا:
مدل آشناری به دلیل ساختار خطی خود، انعطاف‌پذیری بالایی برای تغییرات مداوم ندارد. اما در این پروژه به دلیل پایداری نیازمندی‌ها و شفافیت آن‌ها، نیازی به تغییرات مکرر یا بازبینی‌های پیوسته وجود ندارد. از این رو، مدل آشناری می‌تواند فرآیندی ساده و منظم برای تکمیل پروژه فراهم کند.

پاسخ مسئله‌ی ۳.

بخش اول

در روش اسکرام، یکی از اصول اصلی این است که برنامه‌ریزی اسپرینت، به‌طور مشخص برای یک بازه زمانی کوتاه‌مدت انجام می‌شود و پس از شروع اسپرینت، از تغییر برنامه و افزودن یا حذف آیتم‌ها تا حد ممکن خودداری می‌شود. این اصل به تیم کمک می‌کند که تمرکز خود را حفظ کرده و بتواند به اهداف اسپرینت پایبند بماند. با این حال، برخی شرایط ممکن است نیاز به بازنگری در برنامه‌ریزی را ضروری کند. با در نظر گرفتن شرایط پروژه و ویژگی‌های خاصی که به‌روزرسانی‌های مداوم نیازمندی‌ها ایجاد می‌کند، بهترین رویکرد این است که برنامه‌ریزی اسپرینت فعلی را حفظ کنیم و تغییرات جدید را به اسپرینت‌های آینده موکول کنیم. دلیل این تصمیم:

- حفظ تمرکز بر اهداف فعلی: تیم تنها سه روز از اسپرینت اول را پشت سر گذاشته و در حال حاضر اهداف و کارهای خاصی را برای این دوره تعریف کرده است. تغییر در برنامه اسپرینت در این مرحله ممکن است تمرکز و پیشرفت تیم را مختل کند و باعث کاهش بهره‌وری شود.
- اولویت‌دهی و ارزیابی تغییرات جدید در برنامه‌ریزی بعدی: طبق اصول اسکرام، تیم در انتهای هر اسپرینت جلسه بازنگری و برنامه‌ریزی برای اسپرینت بعدی برگزار می‌کند. با استفاده از این روش، می‌توان به جای ایجاد تغییرات در اسپرینت جاری، قابلیت‌های جدید معرفی شده را در جلسه برنامه‌ریزی اسپرینت بعدی بررسی و اولویت‌بندی کرد و در صورت نیاز، آن‌ها را به فهرست کارهای اسپرینت بعد اضافه کرد.
- توجه به محدودیت زمانی پروژه: با توجه به زمان‌بندی فشرده شش‌ماهه پروژه، تیم باید به‌طور مداوم به اولویت‌بندی نیازمندی‌ها و ویژگی‌ها بپردازد تا تمرکز خود را بر روی وظایف اصلی حفظ کند. انجام تغییرات ناگهانی و غیرضروری در اسپرینت‌های جاری می‌تواند به طولانی شدن پروژه و عدم رعایت مهلت مقرر منجر شود.

بنابراین، با حفظ برنامه‌ریزی اسپرینت فعلی و موکول کردن ارزیابی تغییرات به اسپرینت بعدی، تیم می‌تواند به‌صورت مؤثر و در عین حال انعطاف‌پذیر، قابلیت‌های جدید را در پروژه ادغام کند.

بخش دوم

با توجه به شرایط همکاری با سازمان ملی اسناد و طبیعت پویای پروژه‌ای که نیازمند تطبیق مداوم با نوآوری‌ها و ویژگی‌های جدید در زمینه مدل‌های زبانی است، روش چابک و به‌ویژه اسکرام به عنوان روشگانی مناسب برای مدیریت این پروژه پیشنهاد می‌شود. در ادامه دلایل انتخاب این روش را با توجه به ویژگی‌های خاص پروژه توضیح می‌دهم:

- انعطاف‌پذیری در برابر تغییرات پویا و غیرقابل پیش‌بینی: در این پروژه، نیازمندی‌ها ممکن است به دلیل تغییرات سریع در فناوری مدل‌های زبانی و رقابت شدید در این حوزه به‌طور مداوم تغییر کنند. از آنجا که سما انتظار دارد تمامی قابلیت‌ها و نوآوری‌های رقبا، به‌ویژه قابلیت‌هایی که توسط شرکت‌هایی مانند گوگل و غیره معرفی می‌شوند، پیش از انتشار در محصول خود اعمال شود، لازم است که تیم توسعه توانایی واکنش سریع به تغییرات و تطبیق با شرایط جدید را داشته باشد. روش اسکرام، با تقسیم پروژه به اسپرینت‌های کوتاه و بازبینی نیازمندی‌ها در پایان هر اسپرینت، این انعطاف‌پذیری را فراهم می‌کند و اجازه می‌دهد تا تیم به‌راحتی تغییرات و ویژگی‌های جدید را در برنامه‌ریزی اسپرینت‌های بعدی مدنظر قرار دهد.
- برگزاری جلسات بازنگری و بهبود مستمر:

همکاری نزدیک و هماهنگی با سما به دلیل ماهیت دولتی و نظارت دقیق این نهاد، نیازمند شفافیت و ارزیابی مداوم است. در روش اسکرام، جلسات اسپرینت ریویو (Sprint Review) و رتروسپکتیو (Retrospective) پس از هر اسپرینت به تیم امکان می دهند تا به طور مستمر عملکرد خود را ارزیابی کرده و بر اساس بازخوردهای سما یا نیازهای جدید، مسیر پروژه را بهبود بخشند. این جلسات می توانند به هماهنگی بهتر بین تیم و سما کمک کرده و شفافیت لازم را برای مدیران این نهاد فراهم کنند.

بخش سوم

این اقدامات در دو حوزه اصلی یعنی تست و ارزیابی کیفیت و استقرار و آماده سازی نهایی محصول قرار می گیرند. در ادامه به دو اصل کلیدی برای هر بخش اشاره می کنم.

تست و ارزیابی کیفیت

- اصل تست جامع و چندمرحله ای:

یکی از اصول کلیدی پس از کدنویسی، انجام تست های جامع است. این شامل تست واحد، تست یکپارچه سازی، و تست سیستم است. در تست واحد، هر ماژول و تابع به صورت جداگانه مورد بررسی قرار می گیرد تا اطمینان حاصل شود که به درستی کار می کند. در تست یکپارچه سازی، عملکرد بخش های مختلف سیستم در کنار یکدیگر ارزیابی می شود تا اطمینان حاصل شود که ارتباطات و تعاملات بین آن ها بدون مشکل است. در نهایت، در تست سیستم، محصول به صورت کلی و از دیدگاه کاربر ارزیابی می شود.

- اصل تست پذیرش و تطابق با نیازمندی های مشتری:

پس از انجام تست های داخلی، لازم است تست پذیرش انجام شود تا مطابقت محصول با نیازمندی های تعریف شده مشتری ارزیابی شود. در این مرحله، محصول به صورت آزمایشی و با سناریوهای واقعی شبیه سازی شده بررسی می شود. این مرحله کمک می کند تا اطمینان حاصل شود که محصول نهایی نیازهای خاص سما را برآورده می کند، از جمله امنیت داده ها، قابلیت های مورد انتظار و همچنین تطابق با مقررات خاص.

استقرار و آماده سازی نهایی محصول

- اصل تهیه مستندات جامع و آموزش کاربران:

پیش از تحویل محصول به مشتری، تهیه مستندات جامع برای کاربران و تیم فنی سما ضروری است. این مستندات شامل راهنمای کاربری، آموزش نصب و راه اندازی، و توضیحات فنی مربوط به عملکرد و ویژگی های سیستم است. با ارائه این مستندات و برگزاری جلسات آموزشی، کاربران نهایی و تیم فنی می توانند به راحتی با سیستم کار کنند و از آن بهره برداری کنند.

- اصل آماده سازی زیرساخت های پشتیبانی و خدمات پس از فروش:

یکی از اصول مهم پس از اتمام کدنویسی و پیش از ارائه محصول، فراهم کردن زیرساخت های پشتیبانی و خدمات پس از فروش است. این شامل آماده سازی تیم پشتیبانی، تعیین کانال های ارتباطی برای گزارش مشکلات، و اطمینان از داشتن ابزارهای لازم برای نظارت و رفع اشکالات احتمالی می باشد. این مرحله به سازمان اطمینان می دهد که در صورت بروز مشکلات پس از استقرار، تیم پشتیبانی آماده پاسخگویی به مشکلات و رفع آن ها است.

بخش چهارم

قبل از افزودن این قابلیت به بک لاگ، مراحل زیر را برای ارزیابی دقیق و تصمیم گیری مناسب انجام دهم:

۱. ارزیابی و تحلیل قابلیت پیشنهادی

- بررسی نیاز و ارزش افزوده: ابتدا باید مشخص کنم که این قابلیت پیشنهادی چه ارزش افزوده‌ای برای پروژه به ارمغان می‌آورد و چگونه به اهداف اصلی پروژه کمک می‌کند. از توسعه‌دهنده می‌خواهم توضیحات دقیقی درباره مزایای این قابلیت، تأثیر آن بر تجربه کاربران و نحوه تطبیق آن با نیازهای نهایی سازمان ملی اسناد (سما) ارائه دهد. بررسی این موارد به من کمک می‌کند تا تعیین کنم آیا این قابلیت به افزایش کیفیت و جذابیت محصول نهایی کمک می‌کند یا خیر.
- تحلیل اولویت و تطابق با زمان‌بندی: با توجه به محدودیت زمانی پروژه (بازه شش ماهه)، باید مشخص شود که افزودن این قابلیت چگونه بر برنامه کلی پروژه و اسپرینت‌های آینده تأثیر می‌گذارد. از آنجا که پروژه تحت چارچوب اسکرام مدیریت می‌شود، باید تعیین کنم که این قابلیت چقدر اولویت دارد و آیا می‌توان آن را در اسپرینت‌های آینده جای داد یا خیر. در صورتی که قابلیت پیشنهادی نیاز به تغییر برنامه‌های فعلی داشته باشد، ممکن است افزودن آن به بک لاگ به تأخیر بیفتد.

۲. برگزاری جلسه با تیم و مدیر پروژه

- مشارکت تیم در تصمیم‌گیری: با برگزاری یک جلسه کوتاه با اعضای تیم، نظرات و دیدگاه‌های آن‌ها را در مورد قابلیت پیشنهادی جمع‌آوری می‌کنم. از آنجا که توسعه‌دهندگان دیگر نیز در این پروژه نقش دارند، نظرات آن‌ها می‌تواند به شناسایی چالش‌ها و فرصت‌های احتمالی کمک کند و به تصمیم‌گیری دقیق‌تر کمک کند.
- مشورت با مدیر پروژه یا مالک محصول: تصمیم نهایی درباره افزودن قابلیت جدید باید با هماهنگی مالک محصول یا مدیر پروژه گرفته شود. مالک محصول می‌تواند در تعیین اولویت این قابلیت نسبت به سایر آیتم‌های بک لاگ و ارتباط آن با اهداف کلی پروژه نقش کلیدی داشته باشد. اگر قابلیت پیشنهادی با اهداف و نیازهای اصلی پروژه سازگار باشد، آن را با اولویت مناسب به بک لاگ اضافه می‌کنیم.

۳. افزودن قابلیت به بک لاگ با اولویت‌بندی مناسب

- پس از ارزیابی دقیق و هماهنگی با تیم و مدیر پروژه، در صورت تایید، این قابلیت به بک لاگ پروژه اضافه می‌شود. سپس آن را بر اساس اولویت و تأثیر آن بر پروژه دسته‌بندی کرده و زمان‌بندی مناسبی برای اجرای آن تعیین می‌کنم. اگر این قابلیت از اولویت بالایی برخوردار باشد، می‌توان آن را در اسپرینت‌های آینده در نظر گرفت.

بخش پنجم

۱. عدم تمرکز و طولانی بودن جلسه بازبینی

جلسه بازبینی اسپرینت طبق اصول اسکرام باید کوتاه، منظم و متمرکز بر بازخورد از اسپرینت اخیر و بررسی محصول باشد. با توجه به توضیحات ارائه‌شده، جلسه بازبینی به حدود پنج ساعت کشیده شده و شامل بحث‌های پراکنده‌ای بوده که بیشتر به مشکلات ابزارها و فرآیندهای مدیریتی پرداخته است. این امر نشان می‌دهد که تمرکز جلسه از بازبینی دستاوردهای اسپرینت و ارزیابی محصول منحرف شده و به مسائل جانبی پرداخته شده است. جلسه بازبینی اسپرینت باید بر بازخورد از محصول و آماده‌سازی برای بهبودهای فنی و عملکردی در آینده تمرکز کند و مسائل تیمی و فرآیندهای کاری بهتر است در جلسه بازنگری بررسی شود.

۲. نیاز به مدیریت فشار کاری تیم

موضوع دیگری که در جلسه مطرح شده، گله‌مندی تیم از درخواست اضافه کاری‌های زیاد به دلیل مشکلاتی مانند مریضی یک عضو و نزدیکی به ددلاین بوده است. این امر نشان می‌دهد که توزیع وظایف و برنامه‌ریزی به گونه‌ای انجام نشده که تیم بتواند بدون نیاز به فشار کاری زیاد به اهداف اسپرینت دست یابد. اسکرام به شدت به اهمیت پایداری و تعادل در کار تاکید دارد تا از خستگی و فرسودگی تیم جلوگیری شود. مالک محصول و اسکرام مستر باید به جای درخواست اضافه کاری از اعضای تیم، به بررسی مجدد بک‌لاگ و اولویت‌بندی بهتر وظایف بپردازند و در صورت نیاز، مقیاس یا دامنه کاری را تنظیم کنند. این رویکرد می‌تواند به بهبود انگیزه و بهره‌وری تیم کمک کند و فشار کاری را کاهش دهد.

۳. تداخل میان وظایف اسکرام مستر و مالک محصول

از آنجا که مالک محصول در این اسپرینت درخواست اضافه کاری زیادی داشته، به نظر می‌رسد که تمرکز بیش از حد روی تحویل سریع به جای پایداری و سلامت کاری تیم بوده است. یکی از وظایف اسکرام مستر، حمایت از تیم در برابر درخواست‌های خارج از توان و اطمینان از رعایت اصول چابک در پروژه است. این رویکرد برای حفظ تعادل و سلامت تیم بسیار حیاتی است.