



دانشکده مهندسی
کامپیوتر و فناوری اطلاعات

۱۳۹۹/۷/۱۲



تمرین اول



مهندسی نرم افزار ۲

گروه {۲}

اعضاء گروه:

(۱) محمدرضا اخگری زیری - ۹۶۳۱۰۰۱

(۲) محمدعلی کشاورز - ۹۶۳۱۰۶۱

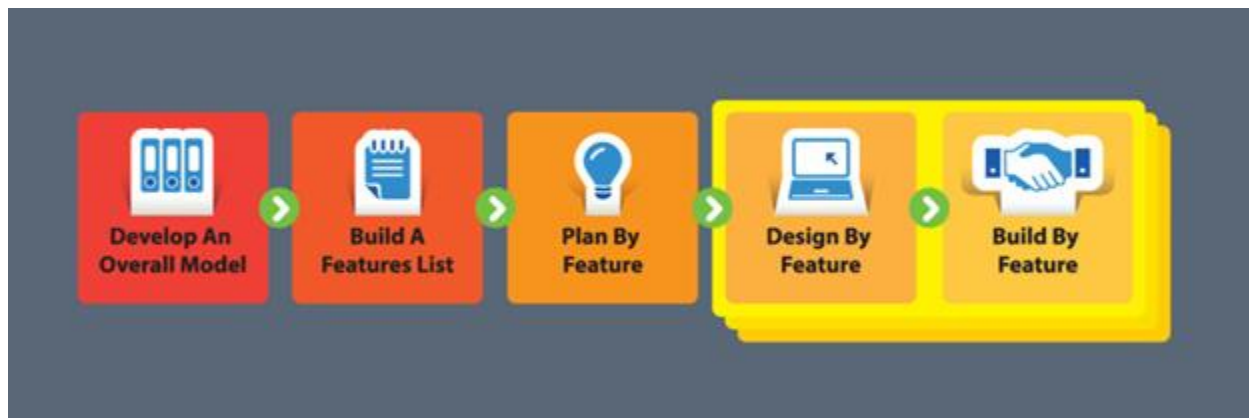
(۳) علی نظری - ۹۶۳۱۰۷۵



۱) پس از مطالعه متدولوژی چابک FDD، به سوالات زیر پاسخ دهید.

الف) میزان و چگونگی مدلسازی در آن را بیان کنید. (مدل‌ها، فازهایی که مدلسازی دارند، میزان مدلسازی، و ...)

¹FDD، فرایندی مبتنی بر معماری و تحویل سریع featureها است. FDD اولین بار در سال ۱۹۹۹ در کتاب Java Modeling in Color with UML معرفی شد. همان‌طور که از نامش مشخص است، feature عنصر اصلی این متدولوژی است. همان‌طور که می‌دانیم، به هر تغییر کوچکی که توسط client² قابلیت تشخیص باشد؛ مثل محاسبه فروش کل، اعتبارسنجی رمز ورود و... را feature می‌نامیم. FDD مانند user story در scrum است. یعنی منبع اصلی شکل‌گیری نیازمندی‌ها همین featureها هستند. همان‌طور که می‌بینیم ۵ فرآیند اصلی در مدل FDD وجود دارد که به صورت تکراری انجام می‌شود.



- مرحله اول:

توسعه مدل کلی است که نتیجه‌ی این مرحله، ایجاد مدل سطح بالا و یادداشت‌ها است. هدف ما در گام اول، شناسایی و درک اصول اولیه‌ای است که در پروژه انجام می‌شود و در طول توسعه، مدل ما باید منعکس‌کننده‌ی فرایند توسعه باشد.

¹ Feature Driven Development

² معادل project stakeholders در agile modeling است.



• مرحله دوم:

ساخت لیست featureها است. باید همه‌ی featureها را لیست کنیم و در گروه‌هایی قرار دهیم که از لحاظ موضوعی به هم مرتبط هستند.

• مرحله سوم:

در مرحله‌ی سوم، featureها را برنامه‌ریزی می‌کنیم تا در مراحل بعدی تیم فنی شروع به پیاده‌سازی آن بکنند. این برنامه‌ریزی متناسب با توانمندی‌های تیم، اهمیت موضوع، میزان ریسک و... می‌باشد.

• مرحله چهارم:

در مرحله چهارم، با هر بار اجرا شدن این مرحله توسط tech-lead تیم، تعداد کمی از فیچرها انتخاب می‌شوند به طوری که به هم مربوط باشند و در طول دو هفته قابل پیاده‌سازی باشند. سپس tech-lead با صاحبان محصولات، نمودار توالی این featureها را استخراج می‌کند و کلاس‌ها و متدهای لازم را تعریف می‌کند و پس از بازبینی به سراغ مرحله‌ی پنجم می‌روند.

• مرحله پنجم:

مرحله‌ی پیاده‌سازی است که در این مرحله کدنویسی انجام می‌شود و بعد از بررسی توسط واحد تضمین کیفیت به محصول نهایی اضافه می‌شود.

مراحل چهارم و پنجم تا پایان همه featureها ادامه دارد؛ به طور کلی سه مرحله اول، در حدود ۲۵ درصد از فرآیند را در بر گرفته و ۷۵ درصد مابقی، در تکرار مراحل چهارم و پنجم است.

برای این که در مراحل چهار و پنج بتوانیم میزان پیشرفت کار را ارزیابی کنیم، همان دو مرحله را به ۶ مرحله تقسیم می‌کنیم؛ سه مرحله اول برای بخش مدل‌سازی بر اساس feature و سه مرحله دوم مربوط به پیاده‌سازی و برنامه‌نویسی است که سهم هر مرحله به شرح زیر است:

Domain Walkthrough	Design	Design Inspection	Code	Code Inspection	Promote To Build
1%	40%	3%	45%	10%	1%



ب) آیا این مقدار مدلسازی، چابکی را نقض نمی‌کند؟ چرا؟ چگونه FDD در دسته متدولوژی های چابک است.

خیر، FDD یک روش چابک است.

یکی از معیارهای اصلی Agile، پیشرفت^۳ مستمر محصول است. در FDD نیز، ما همین رویکرد را دنبال می‌کنیم و این پیشرفت، روی featureها صورت می‌گیرد. مبنای این پیشرفت، به چرخه تکرار دو هفته‌ای بستگی دارد. ویژگی‌ها در عرض این مدت ساخته می‌شوند و هر ویژگی که به زمان ساخت بیشتری نیاز داشته باشد، به featureهای کوچکتری شکسته می‌شود، تا اینکه در چرخه دو هفته‌ای قابل اجرا باشد.

- منابع:

[9 Things You Must Know About FDD – Feature Driven Development](#)

[Feature Driven Development \(FDD\) and Agile Modeling](#)

[Feature-driven development](#)

³ Progress



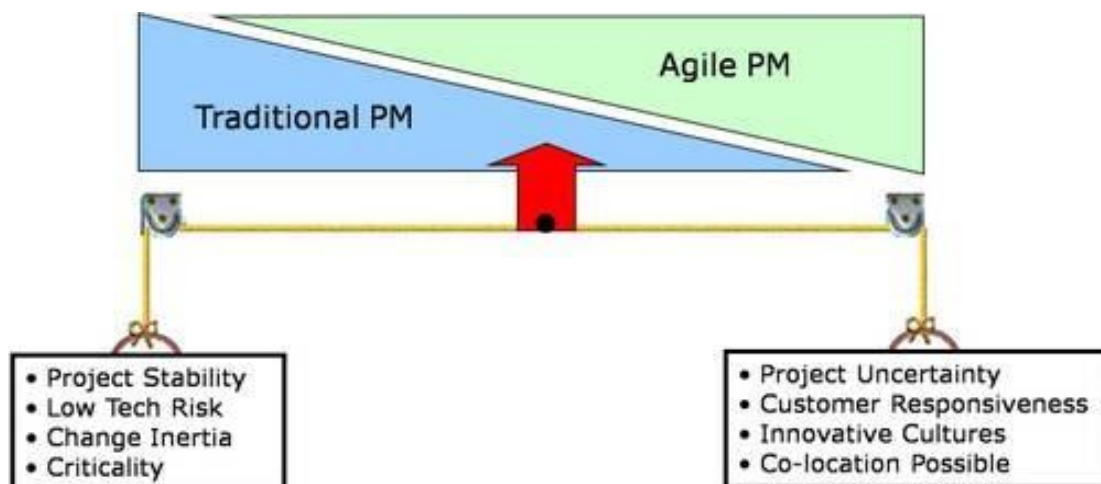
۲) فرایند توسعه مناسب پروژه

الف) درباره فیلترهای سازگاری^۴ مطالعه کرده و این مفهوم را شرح دهید. دو ابزار که از نظر شما کارآمدتر هستند را با ذکر دلیل، انتخاب و معرفی کنید.

فیلترهای سازگاری، ابزارهایی هستند که به ما کمک می کنند تا بفهمیم هر رویکردی در چه شرایطی برای استفاده در پروژهی ما مناسب هستند. این درست نیست که ما در هر شرایطی از متدولوژی چابک استفاده کنیم زیرا ممکن است رویکرد دیگری، تناسب بیشتری با پروژهی ما داشته باشد. برخی از فیلترهای سازگاری را در ادامه معرفی می کنیم:

• The Slider

این فیلتر بیان می کند که انتخاب ما لزوماً نباید مطلق باشد و می تواند به فراخور شرایطی که ما داریم، بخشی از آن به صورت چابک باشد و بخشی از آن این گونه نباشد.



⁴ Suitability Filters



• **DSDM⁵ Suitability Filter**

در این روش با پاسخ دادن به یک سری سوال و ترسیم نموداری بر اساس پاسخ این سوالات، مشخص می‌شود که چه رویکردی برای پروژه‌ی ما مناسب‌تر است. این سوالات حول سه دسته‌بندی کلی به نام‌های فرهنگ^۶، تیم^۷ و پروژه^۸ هستند که در ادامه به اختصار به هر کدام اشاره می‌کنیم:

- **فرهنگ:**

وجود یک فضای حمایتی و اعتماد به تیم که خود این دسته‌بندی شامل سه زیربخش به نام‌های Trust، Buy-In و Decision Making است.

- **تیم:**

این بخش به تعداد اعضای تیم، مهارت و تجربه‌ی هر کدام در پیش‌برد پروژه می‌پردازد و این دسته‌بندی نیز شامل زیربخش‌های Access، Experience و Team Size می‌باشد.

- **پروژه:**

این بخش هم بیشتر ناظر بر میزان تغییرات پروژه است و زیربخش‌های آن شامل Changes، Delivery و Criticality است.

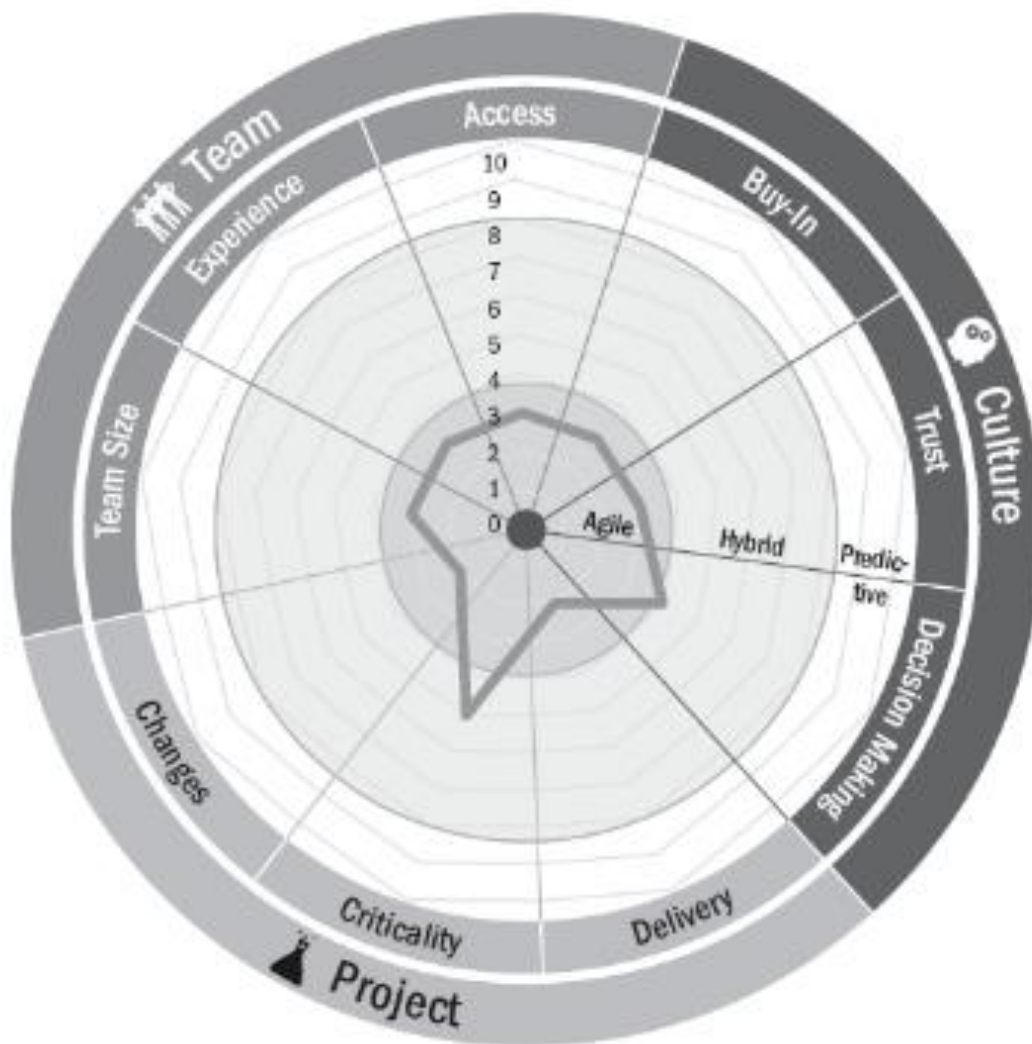
این ۹ زیربخش در کنار هم، نمودار زیر را می‌سازند که به ما در تعیین رویکرد مناسب برای پروژه کمک می‌کند:

⁵ Dynamic Systems Development Method

⁶ Culture

⁷ Team

⁸ Project



با توجه به پاسخی که ما به هر زیربخش داده‌ایم، نمودار بالا برای پروژه‌ی ما تشکیل می‌شود و به ما نشان می‌دهد که بهتر از است از حالت چابک، حالت پیش‌بینی‌شده^۹ و یا اینکه به صورت ترکیبی^{۱۰} باشد.

^۹ Predictive

^{۱۰} Hybrid



• Alistair Cockburn's Criticality and Team Size Factors

این روش دو محور اصلی دارد،

- تعداد اعضای تیم

- حساسیت پروژه.

باید توجه کرد که حساسیت پروژه به این معنی است که در صورت شکست این پروژه نتایج منفی آن به چه صورتی خواهد بود و اهمیت آن چقدر می باشد به این صورت که رویکرد چابک برای پروژه هایی که در صورت شکست عواقب جبران ناپذیر یا سنگینی دارد، توصیه نمی گردد. این روش با کمک این دو معیار، رویکرد مناسب برای پیش برد پروژه به ما پیشنهاد می کند.

• Boehm and Turner – Radar Chart

در این روش، پروژه را بر اساس پنج ویژگی مورد ارزیابی قرار می دهیم و بر اساس نمره ای که به هر کدام از آن ها می دهیم، نموداری را رسم می کنیم که رویکرد مناسب را به ما نشان دهد. در ادامه توضیحی در مورد این ۵ ویژگی می دهیم:

❖ کارکنان^{۱۱}

این ویژگی بیان می کند که مهارت های کارکنانی که روی این پروژه کار می کنند، دارای سطوح مبتدی (سطح ۱) یا حرفه ای (سطح ۲ و ۳) می باشند و با توجه به درصد هر کدام از این سطوح، امتیازی روی نمودار مشخص می شود.

❖ تغییر^{۱۲}

این ویژگی اشاره دارد که احتمال اینکه در پروژه تغییراتی داشته باشیم، چقدر است تا با توجه به آن مشخص شود که کدام رویکرد برای این پروژه مناسب تر است.

❖ فرهنگ^{۱۳}

این ویژگی نیز می خواهد فرهنگ تیم را بررسی کند و مثلاً میزان انعطاف تیم و حتی مدیران در برابر تغییرات را مورد سوال قرار می دهد تا ببینند می توانند با رویکرد چابک ارتباط مناسبی برقرار کنند یا خیر.

¹¹ Personnel

¹² Dynamism

¹³ Culture

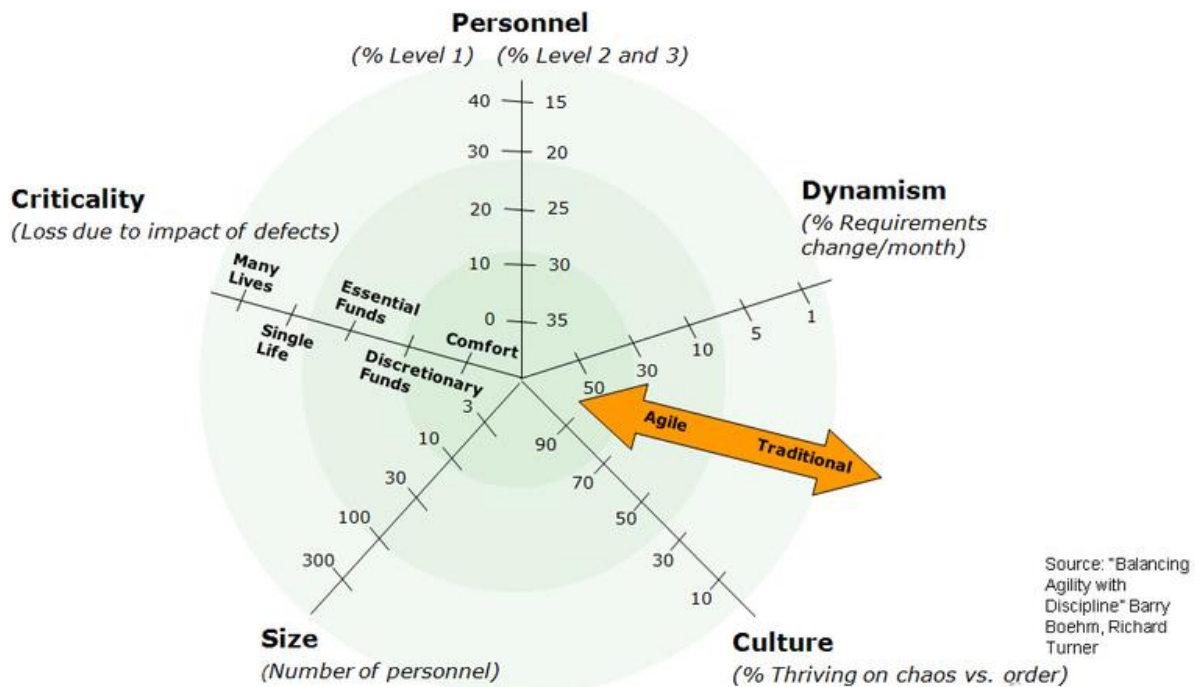


❖ اندازه تیم^{۱۴}

این ویژگی نیز همانند سایر فیلترهای سازگاری که روی اندازه‌ی تیم تمرکز داشتند، رویکرد چابک را برای تیم‌های کوچک‌تر مناسب‌تر می‌داند.

❖ اهمیت و حیاتی بودن^{۱۵}

این ویژگی نیز عاقبت شکست پروژه بررسی می‌کند و رویکرد چابک را برای وقتی که شکست پروژه باعث از دست رفتن منابع کم‌ارزش می‌شود، مناسب‌تر می‌داند.



• Dave Cohen's Agile Factors

در این روش چند پیش‌نیاز برای استفاده از متدولوژی چابک تعیین شده‌است که با توجه به این‌ها تصمیم‌گیری انجام می‌شود. این پیش‌نیازها عبارتند از:

- فرهنگ سازمان باید حمایتی باشد
- افراد باید مورد اطمینان باشند

¹⁴ Team size

¹⁵ Criticality

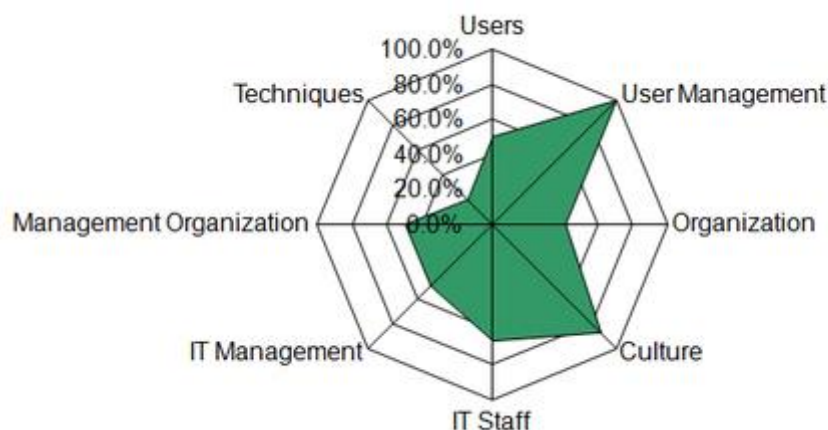


- افراد کم اما توانمند باشند
- تصمیمات تیم‌های برنامه‌نویسی در روند کلی سازمان تاثیرگذار باشد
- سازمان باید بستری داشته باشد تا ارتباطات بین اعضای تیم را تسریع کند

• **The Organizational Suitability Filter**

این روش هم از یک سری سوال از پیش تعیین شده برای مشخص کردن رویکرد مناسب پروژه استفاده می‌کند. تعداد این سوالات در حدود ۳۰ سوال است که در چند بخش قرار می‌گیرند که در زیر به آن‌ها اشاره می‌کنیم:

- مشتری‌های پروژه
- نوع سازمان
- فرهنگ سازمان
- کارکنان
- مدیران IT پروژه
- مدیران سازمان و ذی‌نفعان
- تکنیک‌های مورد استفاده





• The Methodology Bias Hammer

این روش نیز بیان می کند که با در انتخاب یک رویکرد برای پروژه، یک سری مسائل ممکن است انتخاب را به کلی تغییر دهد که تا حدی به خود پروژه ارتباطی ندارند. مثلاً اگر پروژه شما به شدت با رویکرد چابک سنخیت دارد ولی مدیران اصلی پروژه به این رویکرد اعتقادی ندارند پس ممکن است که پروژه به مشکل بخورد پس باید این مسائل در نظر گرفته شود تا ببینیم یک رویکرد آیا با پروژه‌ی ما تناسب دارد یا خیر.

ابزارهای مورد پسند ما:

موارد اول و دوم با هم ترکیب مناسبی را شکل می دهند، زیرا روش The Slider باعث می شود که خودمان را محدود به یک رویکرد نکنیم و در شرایط مختلف ببینیم چه کاری برایمان مفیدتر است و همچنین DSDM Suitability Filter نیز با بهره گیری از ۹ شاخصی که ترتیب می دهد، باعث می شود که انتخاب درستی برای رویکرد کلی پروژه داشته باشیم و اکثر مسائلی که در پروژه دخیل هستند را در نظر بگیریم.



ب) چارچوب کانه وین^{۱۶}، چارچوبی برای حس گیری است که به فهم موقعیت و انتخاب رویکرد مناسب کمک می کند. پس از مطالعه درباره این چارچوب، کارکرد آن را در انتخاب فرایند توسعه مناسب بررسی کنید. به نظر شما نسبت این چارچوب با ابزارهای قسمت قبل چگونه است؟

چارچوب کانه وین ابزاری است که به ما کمک می کند وضعیت موجود را در پنج حالت که ماهیت علت و معلولی دارند، مشخص کنیم تا بتوانیم درک بهتری از وضع موجود داشته باشیم و تصمیمات بهتری بگیریم. در ادامه به این پنج حالت می پردازیم:

آشکار^{۱۷}

در این حالت، ما با شرایطی روبه رو هستیم که مسائل برای همه شفاف هستند و با بررسی مسئله می توانیم به راه حلی درست و مطمئن برسیم، یعنی شرایط در آنها ثابت بوده و در نتیجه راه حل ها نیز معلوم است. به عنوان مثال بانک ها برای وام دادن به مشتریان خود شرایطی را تعیین کرده اند که اگر آن شرایط فراهم شود وام پرداخت می شود. حالا اگر مشتری قسط وام خود را پرداخت نکند (بر اساس تجربیات قبلی) قوانینی وجود دارد که با او برخورد می شود. مشاهده می کنید که در اینجا همه چیز از قبل تعیین شده است و متصدیان بانکی به محض برخورد با این مسائل می توانند آنها را شناسایی و دسته بندی کنند و پاسخ مناسب را ارائه بدهند.

بغرنج^{۱۸}

این محیط ها شامل مواردی است که ما «می دانیم که نمی دانیم». در محیط های سخت و بغرنج راه حل های متنوع و گوناگونی می تواند برای یک مسئله وجود داشته باشد. راه حل هایی که البته مانند محیط های شفاف و ساده از قبل تعیین شده نیستند، بلکه باید با تجزیه تحلیل و آزمون و خطا بدست بیایند. بنابراین فرآیند تصمیم گیری در محیط های سخت به ترتیب شامل درک^{۱۹}، تجزیه^{۲۰} و تحلیل^{۲۱} (به جای دسته بندی در محیط های ساده) و پاسخ است.

¹⁶ Cynefin

¹⁷ Obvious

¹⁸ Complicated

¹⁹ Sense

²⁰ Analyse

²¹ Respond



نمونه‌های این محیط‌های سخت را می‌توان در برخورد با مسائل و چالش‌های مهندسی، وکالت، پزشکان اتاق عمل جراحی، مدیران کسب و کار و... مشاهده کرد که با استفاده از تجربیات خود با موضوعاتی تا حدودی آشنا و تا حدودی ناآشنا برخورد می‌کنند.

پیچیده^{۲۲}

در این حالت ما راه‌حلی برای مسئله نداریم و هم‌چنین اطلاعات کافی هم نداریم پس باید اطلاعات کسب کنیم و سعی کنیم که شرایط را حتی با وجود شکست خوردن بهتر بفهمیم تا بتوانیم در ادامه تصمیم درستی بگیریم.

عامل اصلی قرار گرفتن در این وضعیت نداشتن اطلاعات کافی در مورد مسئله است. مانند زمانی که شرکت‌های بزرگ می‌خواهند برای اولین یک محصول جدید انقلابی را به بازار عرضه کنند که نمونه آن قبلاً وجود نداشته یا بشر با مسائلی روبه‌رو می‌شود که نظیر آن را هرگز ندیده است.

آشفته^{۲۳}

در این حالت ما حتی مسئله را نیز به درستی نمی‌دانیم و هیچ راه حلی هم نمی‌توانیم داشته باشیم و باید به نوعی عمل کنیم که سریع‌تر از حالت آشفتگی بیرون رویم تا بتوانیم با کسب اطلاعات بیشتر مسئله را حل کنیم.

به عنوان مثال‌هایی از این نوع محیط‌ها می‌توان فرماندهان و سربازان در شرایط جنگی، هدایت کشتی در دریای طوفانی، سالم نشانیدن هواپیما در شرایط اضطراری و نقص فنی، مدیریت دارایی در بازارهای مالی پرتلاطم بین‌المللی در شرایط بحران‌های بزرگ و... را نام برد.

اختلال^{۲۴}

این حالت نیز بیانگر وضعیتی است که ما نتوانستیم وضعیت خودمان را در برخورد با این مسئله بفهمیم.

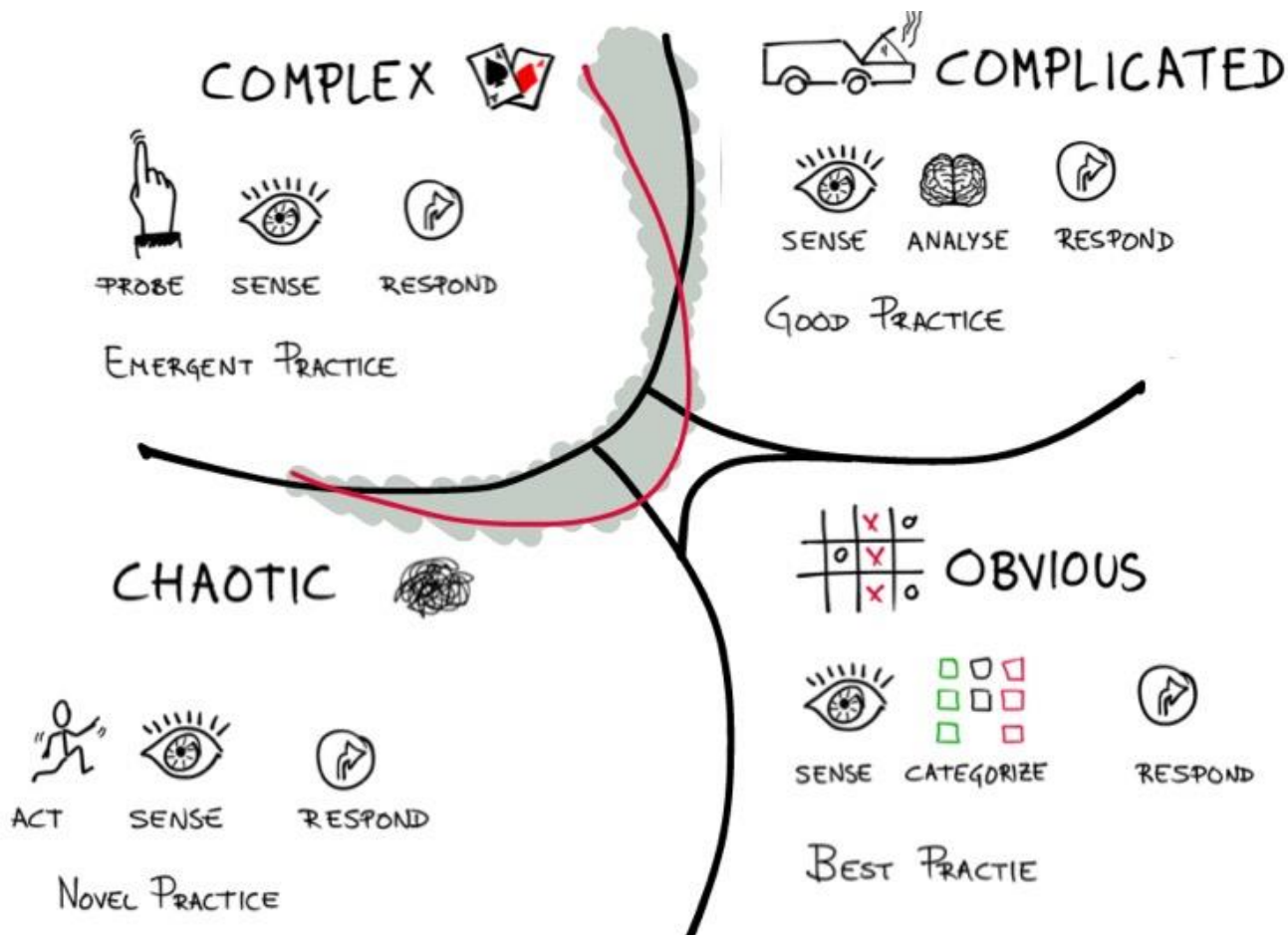
²² Complex

²³ Chaotic

²⁴ Disorder



چارچوب کانهوین نیز به نوعی می تواند یک ابزار برای مشخص کردن رویکرد مناسب برای پروژه‌ی ما باشد. مثلاً وقتی که ما در وضعیت پیچیده هستیم، یعنی مسائل و راه حل آن‌ها کاملاً مشخص نیست و قطعاً نمی توانیم در وضعیت کنونی تمامی موارد را پیش بینی کنیم و از روش سنتی استفاده کنیم پس بهتر است که از رویکرد چابک استفاده کنیم تا روند پیشرفت پروژه به صورت مستمر اتفاق افتد یا مثلاً وقتی که در وضعیت آشکار هستیم یعنی مسائل و راه حل آن‌ها برای ما مشخص و واضح است پس در این جا می توانیم از روش سنتی و پیش بینی شده استفاده کنیم. البته باید انتخاب این رویکرد به صورت جامع تر بررسی گردد اما می بینیم که چارچوب کانهوین نیز دارد در انتخاب رویکرد مناسب به ما کمک می کند.





ج) جایگاه سیستم پروژه گروهتان را در چارچوب کانه وین مشخص کنید.

ما در حالت بغرنج هستیم، زیرا که صورت پروژه برای ما واضح است و همچنین تجربه استفاده از ابزار مشابه باعث شده است تا با قسمت‌های مختلف پروژه آشنا باشیم و وجود موارد مشابه، باعث می‌شود راه‌حل‌های متنوع و گوناگونی برای پیشرفت پروژه پیش روی ما باشد، صرفاً باید برخی از ابهامات درباره نحوه پیاده‌سازی رفع شود.

د) با استفاده از فیلترهای سازگازی و چارچوب کانه وین، فرایند توسعه پروژه گروه خود را مشخص کنید. (با ذکر دلیل)

بر اساس دو فیلتر سازگاری‌ای که ما به عنوان ابزارهای منتخب در قسمت (الف) مشخص کردیم، ما یک تیم سه نفره و جوان هستیم، همچنین همه اعضا به یکدیگر اعتماد کامل داریم و در گذشته با هم کارهای مشترکی انجام داده‌ایم و از روحیه‌ی حمایتی برخورداریم. این پروژه در صورت شکست، عواقب جبران‌ناپذیری برای ما ندارد و ما نیز آماده‌ی ایجاد تغییرات در پروژه هستیم، پس به نظر می‌رسد که با توجه به فیلترهای سازگاری، رویکرد چابک برای پروژه‌ی ما بسیار مناسب‌تر باشد.

همچنین بر اساس چارچوب کانه‌وین در قسمت (ج) دیدیم که پروژه‌ی ما در وضعیت پیچیده قرار دارد و از این لحاظ هم ما نمی‌توانیم همه چیز را از قبل برنامه ریزی و پیش‌بینی کنیم، پس بر اساس چارچوب کانه‌وین نیز رویکرد چابک برای ما مناسب‌تر به نظر می‌رسد.

پس در نهایت با کمک گرفتن از فیلترهای سازگاری و دید شفاف‌ی که با کمک چارچوب کانه‌وین نسبت به پروژه‌ی خودمان پیدا کردیم، تصمیم گرفتیم که در ادامه از رویکرد چابک برای پیش‌برد این پروژه استفاده کنیم.

- منابع:

[کتاب Agile Practice Guide - APPENDIX X3 AGILE SUITABILITY FILTER TOOLS](#)

[Agile suitability filter tools](#)

[Agile Suitability Filters](#)

[چارچوب Cynefin؛ چگونه می‌توانید جلوی تصمیمات اشتباه را بگیرید؟](#)



مدل یا چارچوب کانوین برای تصمیم گیری بهتر

(۳) الگوهای فرآیند

الف) پس از مرور الگوهای فرایند Ambler، برای هر کدام از سطوح تعریف شده، یک الگوی فرآیندی به انتخاب خود، با ذکر منبع بر اساس ساختار و قالب داده شده، ارائه دهید.

آقای Ambler، الگوی فرآیند را به این صورت تعریف کرد که یک دیدگاه و یا یک سری عملیات در عمل ثابت شده، برای ایجاد نرم افزار پیشنهاد می کند.

الگوهای فرایند به سه دسته تقسیم می شوند که به ترتیب عبارتند از:

- کار^{۲۵}:

که جزییات مراحل اجرای یک کار از فرایند را توصیف می کند و ریزدانه ترین الگوی فرایند است.

- مرحله^{۲۶}:

که مراحل لازم برای انجام یک مرحله از فرایند را به تصویر می کشد. هر مرحله معمولاً از یک سری کار تشکیل شده است.

- فاز^{۲۷}:

که تعامل دو یا چند مرحله به منظور انجام یک فاز را به تصویر می کشد و درشت دانه ترین الگوی فرایند است.

فازها در طول فرایند به صورت پشت سر هم و متوالی انجام می شوند ولی مراحل در هر فاز می توانند به صورت تکراری انجام شوند. Ambler برای هر یک از ۳ دسته بالا، الگوهای زیادی پیشنهاد کرده است که برای هر الگو مراحل دقیق و یک سری راهنمایی برای تجمیع الگوها و ایجاد یک فرایند جامع تولید نرم افزار ایجاد کرده است.

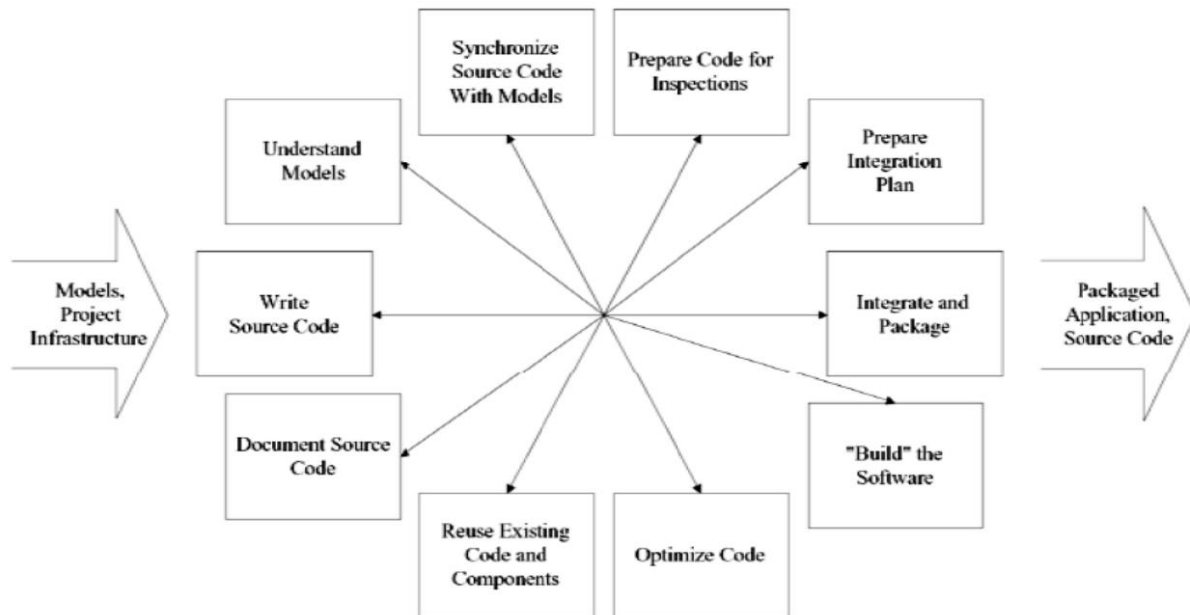
²⁵ Task

²⁶ Stage

²⁷ Phase



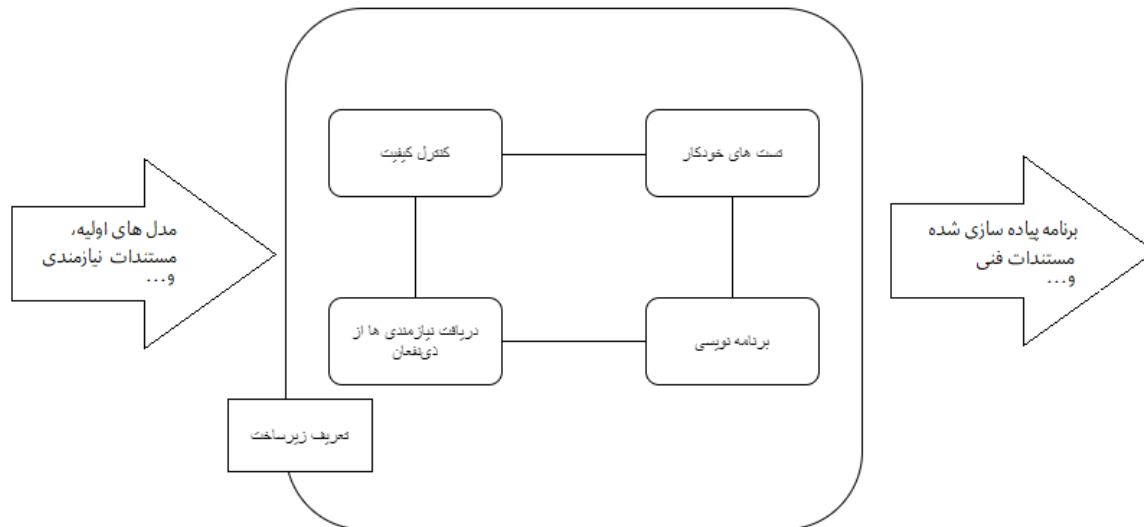
ب) در صورت امکان چند نمونه از استفاده از الگوها در فرآیند پروژه خود را مشخص کنید.
الگوی مرحله و کار را در قالب یک مثال می آوریم:



برای مثال در تصویر بالا همانطور که دیده می شود، مدل ها، معماری و زیرساخت به عنوان ورودی و در نهایت کد برنامه به صورت خروجی است. کل فرایند بالا را از ورودی تا خروجی مرحله می نامند و هر مستطیل در نمودار بالا مانند write source code و ... از آنجایی که قابل شکستن به عنصرهای کوچکتری نیستند، یک الگوی کار است.



برای مثال از الگو فاز می توان به مثال زیر اشاره کرد:



این فاز از چهار مرحله تشکیل شده است که عبارت است از: تست های خودکار که در آن تست ها طراحی و پیاده سازی شده و روی کد اجرا می شوند؛ کنترل کیفیت که در آن افرادی از برنامه استفاده کرده و ایرادات آن را پیدا می کنند؛ برنامه نویسی نیازمندی ها مشخص شده توسط صاحب محصول، پیاده سازی شده است، دریافت نیازمندی ها از ذی نفعان که در طی آن نیازمندی ها از ذی نفعان دریافت و ارزیابی می شود و داکيومنت هایی از آن استخراج می شود.

- منابع:

الگوها و پادالگوها در مهندسی نرم افزار

Process patterns



۴) سازمانی برای انجام پروژه های نرم افزاری خود، تیمی در داخل سازمان جهت استخراج نیازمندی های نرم افزاری، تحلیل و طراحی آن ها دارد. اما به دلیل عدم وجود نیروی متخصص، پیاده سازی، آزمون، و پشتیبانی از نرم افزار را به شرکت هایی در خارج از سازمان خود می سپارد. این رویکرد در تولید و توسعه سیستم ها، “برون سپاری”^{۲۸} نام دارد. پس از مطالعه درباره این رویکرد، به موارد زیر پاسخ دهید (توضیح دهید):

الف) چه مشکلاتی در این روش وجود دارد؟

برون سپاری، علاوه بر مزایا، دارای معایب تکنیکال و مدیریتی نیز هست که گاهی این معایب باعث می شوند تا حتی درمورد خوبی های این روش سخن گفته نشود و به جای آن روش خالص درخانه^{۲۹} استفاده شود [۱]. از معایب این روش می توان موارد زیر را نام برد:

۱. کنترل مدیریت را از دست خواهیم داد [۱] و [۲]
- وقتی شما مدیریت یک بخش را به شرکت دیگری اهدا می کنید، این مسئولیت برعهده آنها گذاشته می شود. شما دیگر قادر به مدیریت مناسب آن شرکت نخواهید بود، آن شرکت که کار را برعهده گرفته است، استانداردهای شما را در نظر نمی گیرد، آن ها فقط به دنبال سودبردن از وظیفه ای که برعهده شان است، هستند.
۲. تغییرات و هزینه های آن ها [۲] و [۳]
- شما در هنگام قرارداد بستن با شرکتی که به آن برون سپاری کردید، باید تمام موارد را به دقت ذکر کنید. در صورت تغییر و یا اضافه شدن درخواست دیگری به شرکت، هزینه های اضافه ای پرداخت خواهد شد.
۳. خطر امنیت و حریم خصوصی [۱]، [۲] و [۳]
- اطلاعاتی که شرکت بر اساس آن جلو می رود، مهم ترین عنصر در شرکت است. اگر شما این اطلاعات را در اختیار شرکت دیگر قرار دهید، امکان افشای آن ها وجود دارد. (باید در قرارداد برای این موضوع تمهیداتی در نظر بگیرید)
۴. مشکلات کیفیت [۱] و [۲]
- شرکتی که کار را به آن برون سپاری می کنید، تنها با پول انگیزه می گیرد. از آنجا که در قراردادی که می بندید، قیمت را قطعی می کنید، تنها راه آن ها برای افزایش سود، کاهش هزینه ها خواهد بود و تنها زمانی که آن ها شرایط قرارداد را برآورده کنند، شما به آنها پول می دهید. علاوه بر این شما توانایی پاسخ دادن به تغییرات سریع محیط

²⁸ Outsourcing

²⁹ pure in-house



کسب و کار را نیز از دست می‌دهید، چرا که قرارداد کاملاً مشخص است و برای هر تغییری شما باید هزینه‌ی اضافه پرداخت کنید.

۵. توان مالی شرکت مسئول [۱]، [۲] و [۳]

شرکتی که شما آن را برای برون سپاری انتخاب کردید، جزوی از محصول شما می‌شود و محصول شما به توان مالی آن شرکت بستگی پیدا خواهد کرد، اگر بنا به دلایلی شرکت، دچار ورشکستگی شود، محصول شما نیمه‌کاره رها خواهد شد.

۶. ذهنیت منفی ایجاد شده [۳]

اگر شما جزو تیمی باشید که برون سپاری باعث ایجاد شغل برایشان شده‌باشد، از این اسم خوشتان خواهد آمد اما اگر برون‌سپاری باعث شده باشد که شغل خود را از دست بدهید، این واژه، برای شما بار منفی خواهد داشت. از سمتی، اگر شما بخشی از کار خود را برون‌سپاری کنید، روحیه‌ی نیروی کارتان ممکن است آسیب ببیند.

۷. مدیریت ارتباطات [۲]

مدیریت ارتباطات در این حالت نسب به حالتی که خالص‌درخانه باشند، با توجه به دور بودن شرکتی که برون سپاری را به عهده گرفته است، سخت‌تر است.

ب) کدام مدل (یا مدل‌های) فرآیندی، به انجام هر چه بهتر چنین پروژه‌هایی کمک می‌کند؟ چگونه؟

با جست‌وجو در منابع مختلف، مشاهده کردیم که مدل‌های plan-driven انتخاب مناسبی برای پروژه‌های برون‌سپاری است، زیرا یکی از مشکلات برون‌سپاری، مقاومت در برابر تغییرات بود. اما این روش‌های plan-driven نیز مشکلاتی دارند، از جمله، مشخص بودن نیازمندی‌ها از ابتدای پروژه، مشاهده نکردن خروجی تا مدتی پس از شروع پروژه و ... از طرفی اگر شرکتی روش چابک داشته باشد، نمی‌تواند از برون‌سپاری استفاده کند و از فواید آن بهره ببرد.

با جست‌وجوهای بیشتر، متوجه شدیم که می‌توان مشکلات روش چابک در برون‌سپاری را تاحدی حل کرد، برای مثال می‌توان با شرکتی همکاری کرد که مدل چابک را اجرا می‌کند، یا برای مشکلات ارتباطات پی‌درپی در روش چابک می‌توان از ایمیل و یا تماس تصویری استفاده کرد.

پس انتخاب ما روشی ترکیبی است، زیرا می‌توان از فواید چابک نیز بهره برد. برای مثال روش افزایشی، با تکرار مراحل فرآیند آشنایی، به ما امکان استفاده از چابک و plan-driven را می‌دهد و در هر مرحله ویژگی‌هایی نیز افزوده می‌شود. این افزایشی بودن به ما امکان محک زدن شرکت را نیز می‌دهد و با توجه به نتیجه‌ی کارشان می‌توان با آن‌ها ادامه همکاری داد.



- منابع:

[Outsourcing Software Applications Development: Issues, Implications, And Impact \(1997\)](#)
[A Comparative Analysis of In-house and Outsourced Development in Software Industry](#)

[برون سپاری چیست و چه مزایا و معایبی دارد؟](#)

- پاسخ تمرین ها را به زبان فارسی و به صورت تایپ شده، در قالب یک فایل Pdf ، در مودل بارگزاری کنید.
- سوالات خود را می توانید از طریق ایمیل از دستیاران تدریس بپرسید.
- فایل پاسخ تمرین را تنها با قالب **SE2-HW1-Group[No].pdf** در مودل بارگزاری کنید.
- بارگزاری تمرین توسط یکی از اعضاء گروه کافی است.
- برای پاسخ های هر قسمت منابع استفاده شده را درج نمایید.
- فایل زیپ ارسال نکنید.
- به ازای هر روز تاخیر در تحویل تمرین ۲۰٪ از نمره تمرین کسر خواهد شد.
- حداقل برخورد به پاسخ های مشابه، تخصیص نمره کامل منفی به طرفین خواهد بود.