مدیریت عصر پیشرفت

تحلیل و بررسی متدولوژی چابک در مدیریت پروژه‌ها: رویکردها، چالش‌ها و کاربردها

شهداد وثوق زاده، دانشجوی دکتری رشته مدیریت فناوری اطلاعات –

مدیریت خدمات و توسعه فناوری اطلاعات، دانشگاه آزاد اسلامی واحد قزوین، قزوین

shahdad.vosoughzadeh11@gmail.com

چکیده

مدیریت پروژه‌ها در دنیای پیچیده و پر تغییر امروز، به‌ویژه در حوزه فناوری اطلاعات و نرم‌افزار با چالش‌های مختلفی مواجه است. از آنجا که نیاز به تحویل سریع، کیفیت بالا، و تطابق با تغییرات مداوم افزایش یافته است، استفاده از روش‌های سنتی مانند مدل آبشاری در بسیاری از پروژه‌ها ناکارآمد به نظر می‌رسد. به همین دلیل، متدولوژی چابک به‌عنوان یک رویکرد نوین در مدیریت پروژه‌ها معرفی شده است که توانسته است در پروژه‌های نرم‌افزاری و سایر صنایع، به بهبود عملکرد و کارایی کمک کند. این مقاله به بررسی متدولوژی چابک، مدل‌های مختلف آن، چالش‌ها و کاربردهای آن در مدیریت پروژه‌ها پرداخته و راهکارهایی برای پیاده‌سازی موفق این متدولوژی در پروژه‌های مختلف ارائه می‌دهد.

این مقاله ابتدا تاریخچه و اصول بنیادین متدولوژی چابک را مرور می‌کند، سپس به تحلیل مدل‌های مختلف چابک شامل اسکرام (Scrum)، کانبان (Kanban) و XP (Extreme Programming) می‌پردازد. علاوه بر این، چالش‌هایی که در پیاده‌سازی متدولوژی چابک وجود دارد، مانند مقاومت در برابر تغییرات، نیاز به تیم‌های ماهر و مشکلات مقیاس‌پذیری، مورد بررسی قرار می‌گیرد. در نهایت، کاربردهای متدولوژی چابک در پروژه‌های نرم‌افزاری و توسعه محصولات جدید در صنایع مختلف تحلیل و پیشنهاداتی برای بهبود فرآیندهای پیاده‌سازی آن ارائه می‌شود.

پژوهش حاضر از نوع مروری-تحلیلی است و بر پایه منابع علمی منتشرشده انجام شده است. یافته‌ها نشان می‌دهد اسکرام پرکاربردترین چارچوب چابک است، کانبان و XP بیشتر با پروژه‌های نرم‌افزاری و کارآفرینی درون‌سازمانی پیوند دارند و رویکردهای ترکیبی به‌عنوان پاسخی به محدودیت‌های چابک در مقیاس بزرگ و محیط‌های مقررات‌محور در حال گسترش هستند. موفقیت پیاده‌سازی بیش از آنکه به انتخاب یک چارچوب خاص وابسته باشد، به حمایت مدیریت، اثربخشی تیم، آموزش، همخوانی فرایند با بستر پروژه و تحول فرهنگی سازمان بستگی دارد.

کلیدواژه‌ها: مدیریت پروژه‌ها، متدولوژی چابک،توسعه محصولات جدید، رویکرد ترکیبی، عوامل حیاتی موفقیت.

۱. مقدمه

با توجه به نیاز روزافزون به پاسخگویی سریع به تغییرات بازار و تقاضای مشتریان، مدل‌های سنتی مدیریت پروژه نظیر مدل آبشاری که در آن پیش‌بینی‌ها و برنامه‌ریزی‌های دقیق از ابتدای پروژه تا انتها انجام می‌شود، قادر به رفع این نیازها نیستند. این مدل‌ها در دنیای امروز که سرعت تغییرات بالا است و پیچیدگی‌های پروژه‌های فناوری اطلاعات روز به روز بیشتر می‌شود، کارآیی کافی ندارند. به‌ویژه در پروژه‌های نرم‌افزاری که نیاز به تعامل مستمر با مشتری و تحویل سریع نسخه‌های اولیه دارند، متدولوژی‌های سنتی نمی‌توانند به‌طور مؤثر کار کنند. (Leech & Hanslo, 2025).

در این شرایط، متدولوژی چابک به‌ویژه در صنایع نرم‌افزار به‌عنوان یک راه‌حل نوین مطرح شده است که بر اساس تعاملات مستمر، تحویل تدریجی و پذیرش تغییرات سریع طراحی شده است. این رویکرد به‌طور مؤثر در پروژه‌هایی که نیاز به انعطاف‌پذیری بالا و پاسخگویی سریع به تغییرات دارند، پیاده‌سازی می‌شود (Barrera & Beltrán, 2026; Suciu et al., 2026).

افزون بر این، پذیرش گسترده چابک به‌تنهایی موفقیت پروژه را تضمین نمی‌کند؛ دستیابی به نتایج پایدار در پروژه‌های چابک همچنان دشوار است و به عوامل حیاتی موفقیت وابسته است (Hanslo & Tanner, 2026). در دنیای پر از تغییرات و پیچیدگی‌های امروزی، پروژه‌ها به‌ویژه در زمینه‌های نرم‌افزاری و فناوری اطلاعات با چالش‌های فراوانی روبه‌رو هستند. پیچیدگی‌های فنی، نیاز به پاسخگویی سریع به تغییرات بازار و نیاز به تحویل سریع محصولات با کیفیت بالا، باعث شده‌اند که روش‌های سنتی مدیریت پروژه قادر به پاسخگویی به این نیازها نباشند. مدل آبشاری که به‌طور عمده بر پیش‌بینی دقیق، برنامه‌ریزی خطی، و انجام هر مرحله از پروژه به‌طور پیوسته تأکید دارد، در پروژه‌های پیچیده و با تغییرات مداوم کارآمد نیست. (Leech & Hanslo, 2025).

در مقابل، متدولوژی چابک که به‌ویژه در پروژه‌های نرم‌افزاری استفاده می‌شود، رویکردی است که بر تغییرات سریع و پذیرش آن‌ها در طول پروژه تأکید دارد. این رویکرد به تیم‌ها این امکان را می‌دهد که در طول پروژه به‌طور مداوم محصولات جدید و بهبود یافته را به مشتریان تحویل دهند و بازخوردهای آن‌ها را برای اصلاحات سریع دریافت کنند. به همین دلیل، متدولوژی چابک به‌ویژه در پروژه‌های نرم‌افزاری و در حال تغییر، به‌عنوان یک رویکرد مؤثر شناخته می‌شود. (Barrera & Beltrán, 2026).

روش پژوهش از نوع مروری-تحلیلی است. منابع آن شامل شش اثر علمی منتشرشده از نوامبر ۲۰۲۵ تا سال ۲۰۲۶ است که متن کامل یا دست‌کم چکیده آن‌ها با شناسه DOI به‌صورت برخط در دسترس است. این منابع شامل سه مرور نظام‌مند، یک مطالعه تجربی با روش آمیخته، یک پیمایش مبتنی بر مدل‌سازی معادلات ساختاری و یک مقاله مفهومی درباره چابکی در مقیاس بزرگ هستند. یافته‌ها بر اساس مضامین تاریخچه و مبانی، مدل‌های چابک، کاربردها، چالش‌ها و عوامل موفقیت دسته‌بندی و تحلیل شده‌اند.

بر این اساس، پرسش‌های اصلی مقاله عبارت‌اند از: متدولوژی چابک بر چه مبانی و چارچوب‌هایی استوار است؟ مهم‌ترین چالش‌های پیاده‌سازی آن کدام‌اند؟ و چه راهکارها و عواملی موفقیت آن را در پروژه‌های گوناگون تقویت می‌کنند؟

۲. مرور ادبیات

۲,۱. تاریخچه و مبانی متدولوژی چابک

متدولوژی چابک ابتدا در دهه ۱۹۹۰ میلادی به‌عنوان پاسخ به محدودیت‌های مدل‌های سنتی مانند مدل آبشاری

معرفی شد. این مدل‌ها به دلیل ساختار خطی و پیش‌بینی‌پذیر خود، در پروژه‌هایی که نیاز به تغییرات مداوم دارند، توانایی لازم را ندارند. در سال ۲۰۰۱، گروهی از متخصصان توسعه نرم‌افزار با انتشار منشور چابک، اصول اولیه این روش را تدوین کردند. این اصول به‌ویژه بر همکاری‌های نزدیک با مشتری، تحویل تدریجی، و پذیرش تغییرات در هر مرحله از پروژه تأکید دارند. (Leech & Hanslo, 2025).

بر پایه مرورهای اخیر، سیر تحول روش‌های مدیریت پروژه در حوزه فناوری اطلاعات از رویکردهای آبشاری به چارچوب‌های چابک و سپس به مدل‌های ترکیبی ادامه یافته است (Leech & Hanslo, 2025). خاستگاه روش‌های چابک به منشور چابک بازمی‌گردد که مجموعه‌ای از اصول راهنمای توسعه نرم‌افزار را معرفی می‌کند (Barrera & Beltrán, 2026). برای نمونه، اسکرام بر ایده‌های ناکاتا و تاکئوچی در دهه ۱۹۸۰ درباره کار تیمی و خودمختاری تیم‌ها استوار است، در دهه ۱۹۹۰ توسط شوابر و ساترلند به‌عنوان روشی برای توسعه نرم‌افزار صورت‌بندی شد و بر منشور چابک، اصول ناب و تجربه‌گرایی تکیه دارد (Barrera & Beltrán, 2026).

۲,۲. مدل‌های مختلف چابک

چندین مدل برای پیاده‌سازی متدولوژی چابک در پروژه‌ها وجود دارد که از جمله آن‌ها می‌توان به مدل‌های اسکرام، کانبان و XP اشاره کرد:

- اسکرام: (Scrum) این مدل یکی از شناخته‌شده‌ترین و موفق‌ترین مدل‌های چابک است که در آن پروژه به فازهای کوتاه‌مدت به نام اسپرینت تقسیم می‌شود. در هر اسپرینت، تیم‌ها به انجام وظایف خاص می‌پردازند و در پایان اسپرینت، نتایج به مشتری تحویل داده می‌شود. اسکرام به‌ویژه برای پروژه‌های نرم‌افزاری که نیاز به تغییرات مداوم دارند، بسیار مناسب است. اسکرام در مرور نظام‌مند Barrera و Beltrán پرتکرارترین روش چابک بوده و در ۹۷ درصد پژوهش‌های بررسی‌شده حضور دارد. (Barrera & Beltrán, 2026).
- کانبان (Kanban): کانبان به‌ویژه برای پروژه‌هایی که نیاز به مدیریت جریان کار دارند، استفاده می‌شود. این مدل از تخته‌های بصری برای پیگیری وضعیت پروژه و مدیریت وظایف استفاده می‌کند. کانبان به تیم‌ها این امکان را می‌دهد که فرآیندهای کاری را بهینه کنند و از آن‌ها برای افزایش کارایی استفاده کنند. این روش بر نظام کششی و خودمدیریتی فرایند، بدون برنامه‌ریزی متمرکز، استوار است. (Barrera & Beltrán, 2026).
- XP (Extreme Programming): این مدل به‌طور خاص برای پروژه‌های پیچیده و با نیاز به کدنویسی با کیفیت بالا طراحی شده است. XP بر توسعه مستمر کد، استفاده از تست‌های خودکار و همکاری مستمر با مشتری تأکید دارد. XP در کنار اسکرام و کانبان از روش‌های رایج چابک در توسعه نرم‌افزار شمرده می‌شود. (Hanslo & Tanner, 2026).

۲,۳. مزایای چابک نسبت به مدل‌های سنتی

متدولوژی چابک به‌ویژه در پروژه‌های نرم‌افزاری و فناوری اطلاعات مزایای قابل توجهی دارد. از جمله این مزایا می‌توان به افزایش تعامل با مشتری، افزایش کیفیت نرم‌افزار از طریق تست‌های مداوم، و کاهش زمان تحویل محصول اشاره کرد. این روش به تیم‌ها این امکان را می‌دهد که در هر مرحله از پروژه به مشتری بازخورد داده

و پروژه را به‌طور مداوم بهبود بخشند. (Barrera & Beltrán, 2026).

یکی از مهم‌ترین مزیت‌های چابک، توانایی تولید خروجی‌های ملموس و عملیاتی پیش از پایان کامل پروژه و انعطاف‌پذیری آن است؛ در مقابل، روش‌های سنتی عموماً با صلبیت و تحویل یک خروجی واحد در پایان اجرا شناخته می‌شوند (Barrera & Beltrán, 2026). با این حال، غلبه اسکرام در پژوهش‌ها را نباید دلیل برتری مطلق آن دانست؛ این غلبه بیشتر بازتاب نهادینه‌شدن و بلوغ آن است و هیچ روش چابکی برای همه بسترهای پروژه و سازمان به یک اندازه مؤثر نیست (Barrera & Beltrán, 2026).

۲,۴. رویکردهای ترکیبی (هیبریدی)

رویکردهای ترکیبی از محدودیت‌های چابک در محیط‌های بزرگ‌مقیاس و مقررات‌محور پدید آمده‌اند و انعطاف تکراری چابک را با حاکمیت ساختارمند درهم می‌آمیزند (Leech & Hanslo, 2025). همچنین گرایش فزاینده‌ای به ترکیب روش‌های چابک دیده می‌شود؛ نمونه شاخص آن «اسکرامبن» (Scrumban) است که مؤلفه‌های اسکرام و کانبان را یکپارچه می‌کند، و کانبان اغلب برای تقویت مزایای هر دو رویکرد با اسکرام تلفیق می‌شود (Barrera & Beltrán, 2026). نتیجه عملی این شواهد آن است که تطبیق با بستر پروژه بر پیروی انعطاف‌ناپذیر از یک چارچوب مقدم است (Leech & Hanslo, 2025).

۳. روش‌شناسی و رویکردهای متدولوژی چابک

۳,۱. فرآیند و ویژگی‌های اصلی متدولوژی چابک

متدولوژی چابک یک رویکرد مدیریت پروژه است که بر تحویل تدریجی، تعامل مستمر با مشتری، و پذیرش تغییرات در طول پروژه تأکید دارد. برخلاف روش‌های سنتی که همه مراحل پروژه را از ابتدا تا انتها پیش‌بینی می‌کنند، چابک رویکردی تکراری و تدریجی دارد که در آن تیم‌ها هر بار نسخه‌ای از محصول را به‌طور جزئی تکمیل کرده و آن را برای بازخورد به مشتری تحویل می‌دهند. این فرآیند، به تیم‌ها این امکان را می‌دهد که در طول پروژه بازخوردهای مستمر از مشتری دریافت کنند و محصول را بر اساس نیازهای تغییر یافته مشتری بهبود دهند.

در متدولوژی چابک، پروژه‌ها به فازهای کوتاه‌مدت یا «اسپرینت» تقسیم می‌شوند. هر اسپرینت معمولاً بین یک تا چهار هفته طول می‌کشد و در انتهای آن، یک نسخه قابل تحویل از محصول ایجاد می‌شود. این نسخه ممکن است تنها شامل ویژگی‌های اولیه باشد، اما مهم این است که هر اسپرینت به‌طور مستمر به پیشرفت پروژه کمک می‌کند. این شیوه امکان می‌دهد محصول کارکردی در طول پروژه و نه فقط در پایان آن به مشتری تحویل شود (Barrera & Beltrán, 2026). ویژگی‌های اساسی فرآیند چابک عبارتند از:

- تکراری بودن (Iterative): تیم‌ها در هر فاز از پروژه، کارهای خود را تکمیل می‌کنند و سپس به ارزیابی آن‌ها می‌پردازند تا در فاز بعدی اصلاحات لازم را اعمال کنند. این فرآیند باعث می‌شود که پروژه به‌طور مستمر بهبود یابد.
- مشتری‌محوری (Customer-Centric): در متدولوژی چابک، تعامل با مشتری از اهمیت ویژه‌ای برخوردار است. تیم‌ها به‌طور مداوم از مشتری بازخورد می‌گیرند و این بازخورد به‌طور مستقیم بر روی ویژگی‌های محصول تأثیر می‌گذارد.
- پذیرش تغییرات (Change Acceptance): در پروژه‌های چابک، تغییرات به‌عنوان بخشی از فرآیند توسعه پذیرفته می‌شوند. این ویژگی

برای پروژه‌هایی که نیاز به انعطاف‌پذیری دارند، بسیار مهم است.

- ارتباطات مستمر (Continuous Communication): در هر فاز از پروژه، اعضای تیم و مشتری باید ارتباطات مستمر داشته باشند تا اطمینان حاصل شود که محصول به‌طور دقیق با نیازهای مشتری تطبیق دارد.

مثال:

در پروژه‌های نرم‌افزاری مانند پروژه‌های Spotify یا Atlassian، تیم‌ها از فرآیند چابک برای طراحی و توسعه ویژگی‌های جدید استفاده می‌کنند. این ویژگی‌ها در قالب اسپرینت‌های کوتاه‌مدت تولید می‌شوند و پس از هر اسپرینت، بازخورد مشتری دریافت شده و اصلاحات لازم در اسپرینت بعدی اعمال می‌شود.

۳,۲. اسکرام (Scrum) به‌عنوان یکی از مدل‌های چابک

اسکرام یکی از مدل‌های شناخته‌شده چابک است که در بسیاری از پروژه‌های نرم‌افزاری و در سازمان‌های بزرگ پیاده‌سازی می‌شود. در این مدل، پروژه به اسپرینت‌های کوتاه‌مدت تقسیم می‌شود که معمولاً بین یک تا چهار هفته طول می‌کشد. در پایان هر اسپرینت، یک نسخه از محصول تولید می‌شود که به مشتری تحویل داده می‌شود. اسکرام دارای پنج عنصر اصلی است که عبارتند از:

- اسپرینت (Sprint): دوره‌های زمانی محدود (معمولاً ۲ تا ۴ هفته) که تیم در آن‌ها بر روی یک مجموعه مشخص از ویژگی‌ها یا وظایف تمرکز می‌کند.
- جلسات روزانه (Daily Scrum): جلسات روزانه که در آن اعضای تیم وضعیت پیشرفت کار خود را بررسی کرده و مشکلات موجود را شناسایی می‌کنند.
- مالک محصول (Product Owner): شخصی که مسئول تعریف و اولویت‌بندی ویژگی‌ها و وظایف پروژه است و اطمینان حاصل می‌کند که نیازهای مشتری در پروژه مدنظر قرار گیرد.
- اسکرام مستر (Scrum Master): شخصی که مسئول نظارت بر اجرای صحیح فرآیند اسکرام است و تیم را در پیاده‌سازی بهترین شیوه‌های چابک راهنمایی می‌کند.
- تیم توسعه (Development Team): گروهی از افراد که وظیفه اجرای اسپرینت‌ها را بر عهده دارند و محصول را مطابق با نیازهای مشتری تولید می‌کنند. (Barrera & Beltrán, 2026).

با وجود این، اسکرام در اصل بر تیم‌های کوچک با اهداف روشن و بازه‌های زمانی کوتاه استوار است. از محدودیت‌های آن می‌توان به کوچک بودن اندازه تیم، تأکید کمتر بر فرایندهای آزمون، فشار ناشی از تحویل سریع و امکان جایگزینی یا افزودن منابع تنها در آغاز اسپرینت اشاره کرد (Barrera & Beltrán, 2026).

مثال:

در شرکت‌های بزرگی مانند Spotify، برخی تیم‌ها از اسکرام برای مدیریت پروژه‌های نرم‌افزاری خود استفاده می‌کنند. هر اسپرینت شامل فعالیت‌های خاصی است که پس از پایان آن، نسخه‌ای از محصول به مشتری تحویل داده می‌شود. این بازخوردها در اسپرینت‌های بعدی به‌کار گرفته می‌شود تا به‌طور مداوم محصول بهبود یابد.

۳,۳. کانبان (Kanban) و مدیریت جریان کار

واژه کانبان در زبان ژاپنی از ترکیب «کان» به معنای بصری و «بان» به معنای کارت یا تخته گرفته شده است. کانبان یک روش مدیریتی مبتنی بر نظام کششی است که

در آن هر مرحله از فرایند تنها همان چیزی را تولید یا منتقل می‌کند که مرحله بعدی نیاز دارد (Barrera & Beltrán, 2026). برخلاف اسکرام، کانبان الزاماً بازه‌های زمانی ثابت ندارد و بر نمایش بصری کارها و بهینه‌سازی جریان تمرکز می‌کند. در پژوهش‌های بررسی‌شده، کانبان هم در پروژه‌های نرم‌افزاری و هم در سایر پروژه‌های سازمانی به‌کار رفته و اغلب با اسکرام تلفیق شده است (Barrera & Beltrán, 2026).

۳,۴. برنامه‌نویسی فرین (XP) و کیفیت فنی

XP بر کیفیت فنی محصول تأکید دارد و به‌طور معمول با تکرارهای کوتاه، توسعه آزمون‌محور، برنامه‌نویسی دونفره، بازآرایی کد و یکپارچه‌سازی مستمر همراه است. بر پایه مرور Barrera و Beltrán، XP بیشتر در پژوهش‌های مربوط به پروژه‌های نرم‌افزاری و کارآفرینی درون‌سازمانی دیده می‌شود (Barrera & Beltrán, 2026). از نگاه تحلیلی، XP را می‌توان مکمل اسکرام دانست؛ زیرا اسکرام ساختار مدیریتی تیم را فراهم می‌کند و XP بر شیوه‌های مهندسی نرم‌افزار تمرکز دارد.

۳,۵. مقایسه تطبیقی چارچوب‌های اصلی

جدول ۱ ویژگی‌های اصلی سه چارچوب را مقایسه می‌کند.

جدول ۱. مقایسه تطبیقی اسکرام، کانبان و XP

ویژگی	اسکرام	کانبان	XP
ساختار زمانی	اسپرینت‌های محدود، معمولاً یک تا چهار هفته	جریان پیوسته بدون تکرار ثابت	تکرارهای کوتاه
نقش‌ها	مالک محصول، اسکرام مستر، تیم توسعه	نقش الزامی از پیش تعیین‌شده ندارد	برنامه‌نویسان، مشتری و مربی
ابزار و رویه‌های اصلی	اسپرینت، جلسه روزانه، فهرست کارهای محصول	تخته بصری و نظام کششی	آزمون‌محوری، برنامه‌نویسی دونفره، یکپارچه‌سازی مستمر
رویکرد به تغییر	تغییر عمدتاً در آغاز اسپرینت بعدی	تغییر با رعایت ظرفیت جریان کار	پذیرش تغییر در هر تکرار
تمرکز اصلی	مدیریت محصول و هماهنگی تیم	بهینه‌سازی جریان کار	کیفیت فنی کد
جایگاه در پژوهش‌ها	پرتکرارترین روش؛ حضور در ۹۷ درصد پژوهش‌ها	اغلب در تلفیق با اسکرام	عمدتاً در پروژه‌های نرم‌افزاری و کارآفرینی درون‌سازمانی

منبع: تحلیل نگارنده بر پایه (Barrera & Beltrán, 2026; Hanslo & Tanner, 2026)

۴. چالش‌ها و مشکلات متدولوژی چابک

۴,۱. مقاومت در برابر تغییر

مقاومت در برابر تغییر یکی از اصلی‌ترین چالش‌های پیاده‌سازی متدولوژی چابک است. بسیاری از سازمان‌ها به روش‌های سنتی مدیریت پروژه عادت کرده‌اند و تغییر به سمت چابک ممکن است برای آن‌ها دشوار باشد. این مقاومت ممکن است به‌ویژه در میان مدیران ارشد و اعضای تیم که به روش‌های ثابت و پیش‌بینی‌شده عادت کرده‌اند، مشاهده شود.

برای مقابله با این چالش، نیاز به آموزش‌های مؤثر، پشتیبانی مدیران ارشد و ایجاد انگیزه در کارکنان برای پذیرش تغییرات است. از آنجا که متدولوژی چابک نیازمند تغییرات فرهنگی در سازمان‌ها است، این فرایند ممکن است زمان‌بر و دشوار باشد. (Suciu et al., 2026).

پژوهش‌های تجربی اخیر این چالش را دقیق‌تر توضیح می‌دهند. مطالعه Suciu و همکاران (۲۰۲۶) با روش آمیخته و نظریه داده‌بنیاد نشان داد موانع پذیرش چابک اغلب از علت‌های حل‌نشده برمی‌خیزند و با گذشت زمان آن علت‌ها را تقویت می‌کنند؛ این موانع در پنج دسته اصلی جای می‌گیرند و کاهش آن‌ها هم‌زمان به بهبود فرایندها و تحول فرهنگی نیاز دارد. همچنین پیمایش Kucharska (۲۰۲۶) بر ۱۸۷۱ کارمند دانشی نشان داد ادراک کارکنان با ذهنیت چابک و غیرچابک از یکدیگر متقابلاً ناسازگار است و همین موضوع همکاری را دشوار می‌کند، در حالی که فرهنگ چابک به هر دو گروه، به‌ویژه گروه چابک‌اندیش، سود می‌رساند و می‌تواند این تنش‌ها را کاهش دهد.

راهکار:

برای کاهش مقاومت در برابر تغییر، پیشنهاد می‌شود که سازمان‌ها در ابتدا به‌طور آزمایشی از چابک در برخی پروژه‌های کوچک استفاده کنند و نتایج آن را به سایر بخش‌های سازمان گسترش دهند. همچنین، آموزش مستمر و آگاه‌سازی مدیران در مورد مزایای متدولوژی چابک می‌تواند مؤثر باشد. (Kucharska, 2026; Leech & Hanslo, 2025).

۴،۲. نیاز به تیم‌های با تجربه

یکی دیگر از چالش‌ها، نیاز به تیم‌های ماهر و باتجربه است. تیم‌های فاقد تجربه در استفاده از روش‌های چابک ممکن است نتوانند به‌طور مؤثر از این رویکرد استفاده کنند. این مشکل می‌تواند به کاهش کیفیت و سرعت پروژه‌ها منجر شود. (Hanslo & Tanner, 2026; Suciu et al., 2026).

مرور نظام‌مند Hanslo و Tanner (۲۰۲۶) که بر ۵۳ مطالعه اولیه استوار است، ۲۱ عامل حیاتی موفقیت را شناسایی کرد و نشان داد اثربخشی تیم و مدیریت پروژه بیش از سایر عوامل در پژوهش‌ها تکرار شده‌اند. این یافته با تأکید Suciu و همکاران (۲۰۲۶) بر نقش عوامل انسانی در موفقیت پذیرش چابک همسو است.

راهکار:

یکی از راهکارهای مؤثر برای حل این مشکل، سرمایه‌گذاری در آموزش‌های حرفه‌ای و فراهم کردن دوره‌های آموزشی مرتبط با چابک برای اعضای تیم است. همچنین، تیم‌ها می‌توانند از مشاوران خبره در این زمینه بهره‌برداری کنند تا راهنمایی‌های لازم را دریافت کنند. (Hanslo & Tanner, 2026).

۴،۳. مقیاس‌پذیری در پروژه‌های بزرگ

در پروژه‌های بزرگ و پیچیده، مقیاس‌پذیری می‌تواند مشکل‌ساز باشد. نیاز به هماهنگی دقیق میان تیم‌ها، یکپارچگی فرآیندها و استفاده از ابزارهای مدیریتی مناسب برای مقیاس‌بندی متدولوژی چابک امری ضروری است. (Britto et al., 2026).

شواهد اخیر نشان می‌دهد چارچوب‌های چابک بزرگ‌مقیاس همچنان عمدتاً انسان‌محور و دستی‌اند و به جلسات هماهنگی، همگام‌سازی مستندات و واگذاری‌های نقش‌محور تکیه می‌کنند که تطبیق لحظه‌ای را محدود می‌کند (Britto et al., 2026). از این‌رو مرور Barrera و Beltrán (۲۰۲۶) تأکید می‌کند که کارایی روش‌های چابک به‌تنهایی در پروژه‌های بزرگ هنوز به‌صورت بهینه نشان داده نشده و ادغام اصول چابک با روش‌های سنتی مسیر پژوهشی مهمی است. مرور Leech و Hanslo (۲۰۲۵) نیز پیدایش مدل‌های ترکیبی را پاسخی به همین محدودیت‌ها معرفی می‌کند.

راهکار:

برای حل این مشکل، سازمان‌ها باید ابزارهای مدیریت پروژه پیشرفته‌ای را برای تسهیل ارتباطات بین تیم‌ها و نظارت بر پیشرفت پروژه در مقیاس بزرگ استفاده کنند. همچنین، پیاده‌سازی مدل‌هایی مانند Scaled Agile

Framework (SAFe) که برای پروژه‌های بزرگ طراحی شده است، می‌تواند به تسهیل مقیاس‌پذیری کمک کند. (Britto et al., 2026). با این حال، انتخاب چارچوب مقیاس‌پذیر باید با بستر سازمان و الزامات حاکمیتی آن تطبیق داده شود (Leech & Hanslo, 2025).

در افق پژوهشی، «مانیفست چابک بزرگ‌مقیاس بومی هوش مصنوعی» پیشنهاد می‌کند هوش مصنوعی به‌جای ابزاری جانبی، مشارکت‌کننده‌ای اصلی در سازمان‌دهی توسعه بزرگ‌مقیاس باشد و بر شش اصل شامل فرایندهای موازی، تیم‌های نیت‌محور، دانش زنده، اطمینان مبتنی بر راستی‌آزمایی، نیروی کار عامل‌های هماهنگ‌شده و نقشه‌های بازاستفاده‌پذیر استوار است (Britto et al., 2026).

۵. کاربردهای متدولوژی چابک در صنایع و انواع پروژه‌ها

اگرچه چابک ریشه در توسعه نرم‌افزار دارد، کاربرد آن از این حوزه فراتر رفته است. مرور نظام‌مند Barrera و Beltrán (۲۰۲۶) که ۸۰ پژوهش گزینش‌شده از میان ۳۶۷ سند نمایه‌شده در پایگاه Scopus را تحلیل کرده، نشان می‌دهد ۴۱٫۲۵ درصد پژوهش‌ها به توسعه نرم‌افزار و ۵۸٫۷۵ درصد به سایر انواع پروژه اختصاص دارد. جدول ۲ توزیع پژوهش‌ها را بر حسب نوع پروژه نشان می‌دهد.

جدول ۲. توزیع پژوهش‌های بررسی‌شده بر حسب نوع پروژه

نوع پروژه	تعداد	سهم (درصد)
توسعه نرم‌افزار	۳۳	۴۱٫۲۵
کارآفرینی درون‌سازمانی	۱۴	۱۷٫۵
هر نوع پروژه	۹	۱۱٫۲۵
آموزشی و دانشگاهی	۸	۱۰
ساخت‌وساز	۴	۵
فناوری اطلاعات	۳	۳٫۷۵
غیرنرم‌افزاری	۳	۳٫۷۵
پژوهشی	۲	۲٫۵
علم داده	۱	۱٫۲۵
نامشخص	۳	۳٫۷۵
جمع	**۸۰**	**۱۰۰**

منبع: (Barrera & Beltrán, 2026)

در پروژه‌های نرم‌افزاری، اسکرام در بیشتر پژوهش‌ها به‌کار رفته و کانبان و XP در شمار کمتری از آن‌ها دیده می‌شوند. در پروژه‌های کارآفرینی درون‌سازمانی، که ابتکارهای نوآورانه در درون سازمان‌های موجود را در بر می‌گیرد، اسکرام در کنار کانبان، Lean و Crystal گزارش شده است. اسکرام در پروژه‌های ساخت‌وساز، پژوهشی، علم داده و آموزشی نیز، از جمله پروژه‌های کلاسی، به‌کار رفته است (Barrera & Beltrán, 2026). محور اصلی مسئله در پژوهش‌های بررسی‌شده، خود کاربرد روش چابک (۵۶٫۲۵ درصد) و پس از آن کارایی پروژه (۱۷٫۵ درصد) و آموزش (۱۱٫۲۵ درصد) بوده است (Barrera & Beltrán, 2026). این یافته‌ها نشان می‌دهد چابک را می‌توان تقریباً برای هر پروژه‌ای با اهداف روشن به‌کار گرفت، هرچند تناسب آن با پروژه‌های بزرگ‌مقیاس هنوز نیازمند آزمون است.

۶. عوامل حیاتی موفقیت و پیشنهادهایی برای پیاده‌سازی

مرور Hanslo و Tanner (۲۰۲۶) نشان می‌دهد دستیابی به موفقیت پایدار در پروژه‌های چابک دشوار است و موفقیت به مجموعه‌ای از عوامل حیاتی وابسته است که اثربخشی تیم و مدیریت پروژه در صدر آن‌ها قرار دارند. مرور Leech و Hanslo (۲۰۲۵) نیز حمایت رهبری، یکپارچه‌سازی متناسب فرایند و سازوکارهای بهبود مستمر را شرط موفقیت می‌داند. با ترکیب این یافته‌ها با

چالش‌های بخش ۴، پیشنهادهای زیر برای پیاده‌سازی موفق ارائه می‌شود:

- آغاز آزمایشی در پروژه‌های کوچک و گسترش تدریجی نتایج به سایر بخش‌های سازمان.
- حمایت فعال مدیران ارشد و آگاه‌سازی آنان از مزایا و الزامات چابک (Leech & Hanslo, 2025).
- سرمایه‌گذاری در آموزش و ارتقای اثربخشی تیم‌ها (Hanslo & Tanner, 2026).
- اصلاح هم‌زمان فرایندها و فرهنگ سازمانی، زیرا کاهش موانع به هر دو نیاز دارد (Suciu et al., 2026).
- ایجاد فرهنگ چابک و مدیریت تنش میان کارکنان با ذهنیت چابک و غیرچابک (Kucharska, 2026).
- انتخاب یا ترکیب چارچوب بر اساس نوع پروژه و بستر سازمان، به‌جای پیروی انعطاف‌ناپذیر از یک روش (Barrera & Beltrán, 2026; Leech & Hanslo, 2025).
- در پروژه‌های بزرگ، استفاده از چارچوب‌های مقیاس‌پذیر و ابزارهای هماهنگی و پایش پیشرفت (Britto et al., 2026).

نتیجه‌گیری پژوهش

در این مقاله، متدولوژی چابک به‌ویژه در پروژه‌های نرم‌افزاری و توسعه محصولات جدید تحلیل شد. چابک با تأکید بر انعطاف‌پذیری، تعامل مستمر با مشتری و پذیرش تغییرات در طول پروژه، می‌تواند به‌طور مؤثر در پروژه‌های پیچیده، سریع و در حال تغییر به کار گرفته شود. با این حال، چالش‌هایی از جمله مقاومت در برابر تغییر، نیاز به تیم‌های ماهر و مشکلات مقیاس‌پذیری وجود دارد که در پیاده‌سازی این متدولوژی باید مورد توجه قرار گیرند. برای پیاده‌سازی موفق چابک، توجه به آموزش‌های لازم، پشتیبانی از مدیریت، و استفاده از ابزارهای مناسب مدیریتی ضروری است.

بررسی منابع اخیر نشان داد اسکرام پرکاربردترین چارچوب است، اما این برتری آماری به معنای برتری مطلق نیست؛ کانبان و XP نقش مکمل دارند و رویکردهای ترکیبی در محیط‌های بزرگ‌مقیاس و مقررات‌محور اهمیت می‌یابند. موفقیت پیاده‌سازی بیش از هر چیز به عوامل انسانی و سازمانی، یعنی حمایت رهبری، اثربخشی تیم، آموزش و فرهنگ چابک، وابسته است.

این پژوهش محدودیت‌هایی دارد. این پژوهش یک پژوهش تحلیل است پیشنهاد می‌شود پژوهش‌های بعدی اثر رویکردهای ترکیبی را بر کارایی پروژه‌ها بسنجند، ادغام اصول چابک در روش‌های سنتی برای پروژه‌های بزرگ را بررسی کنند و نقش هوش مصنوعی را در چابکی بزرگ‌مقیاس به‌صورت تجربی ارزیابی نمایند.

منابع:

Barrera, J. A., & Beltrán, J. E. (2026). Project management and agile methodologies – systematic review and bibliometric analysis. Ad-Gnosis, 15(17), e-1114. https://doi.org/10.21803/adgnosis.1114

Britto, R., Palmgren, F., Saini, N., & Ohlin, M. (2026). The AI-native large-scale agile software development manifesto. arXiv. https://doi.org/10.48550/arXiv.2605.07717

Hanslo, R., & Tanner, M. (2026). A framework of critical success factors for agile software development. In Proceedings of the 41st ACM/SIGAPP Symposium on Applied Computing (SAC '26) (pp. 986–993). ACM. https://doi.org/10.1145/3748522.3779989

Suciu, D. M., Motogna, S., & Petrescu, M. A. (2026). An empirical concept model of difficulties in agile adoption. In J. F. Krems, H. P. da Silva, & P. Cipresso (Eds.), Computer-Human Interaction Research and Applications (CHIRA 2025) (pp. 484–499). Springer. https://doi.org/10.1007/978-3-032-16448-3_27

Kucharska, W. (2026). How do the internal dynamics of agile and non-agile mindset employees, influenced by agile culture, impact organizational change management? Journal of Organizational Change Management, 39(8). https://doi.org/10.1108/JOCM-12-2025-1093

Leech, B., & Hanslo, R. (2025). The evolution of agile and hybrid project management methodologies: A systematic literature review. arXiv. https://doi.org/10.48550/arXiv.2511.02859

Analysis and Review of Agile Methodology in Project Management: Approaches, Challenges, and Applications

Shahdad Vosoughzadeh, PhD Candidate in Information Technology Management – IT Services Management and Development, Islamic Azad University, Qazvin Branch, Qazvin, Iran

Abstract

Project management in information technology is increasingly challenged by rapid change and by demands for fast delivery and high quality, which limit the suitability of traditional waterfall approaches. This analytical review examines agile methodology, its main frameworks (Scrum, Kanban and Extreme Programming), implementation challenges, applications and success factors, drawing on six scholarly sources published between November 2025 and 2026 that are accessible online with DOIs. The findings indicate that Scrum is the most widely studied framework, that Kanban and XP are mainly associated with software and intrapreneurial projects, and that hybrid approaches are emerging in response to the limitations of agile in large-scale and regulated environments. Resistance to change, the need for skilled teams and scalability remain the principal barriers. Successful adoption depends on leadership support, team effectiveness, training, tailored process integration, continuous improvement and cultural transformation. Practical recommendations for a staged, context-sensitive implementation are presented.

Keywords: project management, agile methodology, hybrid approaches, critical success factors