



1





بررسی ویژگی الگوهای معماری نرم

افزار و ارزیابی و مقایسه متدولوژی


0000

فرایند چابک و (فرایند

عمومی SDLC=System

00000

[2pro.ir](http://2pro.ir)  
(Development Life Cycle



# الگوهای فرایند برای توسعه نرم افزار مبتنی بر مؤلفه



## Process Patterns for Component-Based Software Development

---

Ambler, S. W., The Agile System Development Lifecycle (SDLC), Published on the Web at: [1]  
<http://www.ambyssoft.com/essays/agileLifecycle.html>, 2006, visited in: November 2007

Ramsin R., The Engineering of an Object-Oriented Software Development Methodology. PhD thesis, Y. ORK. UK. [2]  
.Department of Computer Science. April 2006

Ambler, S. W., More Process Patterns: Delivering Large-Scale Systems Using Object Technology. Cambridge University [3]  
.Press, 1999

Craddock, A., DSDM and Scrum: FAQ's - The similarities, differences and potential inter-operability issues, published on [4]  
the Web at  
<http://www.dsdm.org/news/article.asp?id=27>, 2006, visited in: December 2007

R. A. Jeff Garland, Large-Scale Software Architecture, 2003 [5]

L. B. Paul Clements, Rick Kazman, Software Architecture in Practice, Addison Wesley, April 2003 [6]



الگو چیست؟  
 ایجاد الگو  
 ویژگی الگوهای معماری الگو  
 تاریخچه‌ای از الگو  
 تعاریفی از الگوها  
 نرم افزارهای متدولوژی‌های چابک Agile

مقایسه متدولوژی فرایند چابک و (فرایند عمومی SDLC=System Methodologies)

ضعف‌های الگوهای عمومی (Development Life Cycle) فرایند فرایند چابک عمومی  
 بهبود یافته  
 Patterns

طبقه‌بندی برای الگوهای فرایندی Process

الگوهای نتیجه‌گیر (Patterns) فرایند فاز Process Phase Pattern

## الگو چیست ؟



- الگوها مجموعه‌ای از تجارب مهندسين ماهر نرم‌افزار را شامل مي‌شوند
- الگوها تجارب موفق اثبات شده در توسعه نرم‌افزار را ضبط مي‌کنند و به شما کمک مي‌کنند که تکنیک‌های طراحی‌تان را افزایش دهید و به‌دین مورد يك مسأله خاص تکرار شونده در طراحی و یا پیاده‌سازی سیستم نرم‌افزاری بحث مي‌کند
- الگوها مي‌توانند معماری نرم‌افزار را با ویژگی‌های خاصی بوجود آورند
- الگوها دارای راه حل‌هایی برای بسیاری از مشکلات طراحی هستند که ممکن است دوباره ظاهر شوند

در اغلب حوزه‌ها مانند معماری، اقتصاد، و مهندسی نرم‌افزار نوعی (رفتارخبره Expert Behavior) فکرکردن به جفت (مسأله- راه حل Problem-Solution)، وجود دارد، این یک روش طبیعی برای کنار آمدن با هر نوع مسأله یا تعامل اجتماعی است

❖ توجه به نوع مسأله و راه حل آن و در نظرگرفتن برخی فاکتورها منجر به ایجاد الگوها می‌شود

❖ جفت (Problem-Solution) منجر به تشکیل خانواده‌ای از مسائل مشابه و راه‌حلهایی برای هر خانواده می‌شود که این نشانگر یک الگو از مسائل و راه‌حل‌ها است

# تاریخچه الگوها

در اوایل دهه 1990، افراد زیادی روی الگوهای طراحی کار می‌کردند، اما در اواخر 1994 چهار نفر به نام های، گاما، جاکوبسون، هلم و ولسایدز بیشترین تأثیر را با انتشار کتابی به نام:

"Design Pattern: Elements of Reusable Object-Oriented Software"، داشتند.

این چهار نویسنده به Gang-Of-Four = GOF مشهور شدند و بعضی به آنها گروه گاما هم می‌گویند. آنها در این کتاب ایده استفاده از الگوها را در طراحی نرم‌افزار به کار بردند. و یک فرمت استاندارد را برای مستندسازی الگوها ایجاد کرد (الگوهای نوع و ساختار الگوها و الگوهای سازنده الگوها) (Creational Pattern)

همه الگوها که در این دسته قرار می‌گیرند در ارتباط با روش‌های ایجاد اشیاء هستند

◆ (الگوهای ساختاری Structural Pattern)

این نوع الگوها شرح می‌دهند چگونه اشیاء و کلاس‌ها می‌توانند

در ساختارهای بزرگتر باهم ترکیب شوند  
◆ (الگوهای رفتاری Behavioral Pattern)

این نوع الگوها روی ارتباط اشیاء با یکدیگر تمرکز دارند



# ..... تقسیم بندی الگو طراحی \_ گروه گاما



Creational	Structural	Behavioral
<p><b>Factory Method</b>  <b>Abstract Factory</b>  <b>Builder</b>  <b>Prototype</b>  <b>Singleton</b></p>	<p><b>Bridge</b>  <b>Composite</b>  <b>Decorator</b>  <b>Flyweight</b>  <b>Façade</b>  <b>Proxy</b></p>	<p><b>Interpreter</b>  <b>Template Method</b>  <b>Chain of Responsibility</b>  <b>Command</b>  <b>Iterator</b>  <b>Mediator</b>  <b>Memento</b>  <b>Observer</b>  <b>State</b>  <b>Strategy</b>  <b>Visitor</b></p>
<p>الگوهای                      بوجود آورنده</p>	<p>الگوهای                      ساختاری</p>	<p>الگوهای                      رفتاری</p>

Christopher Alexander در کتاب *The Timeless Way of Building* می‌نویسد:

❖ هر الگو یک قاعده سه بخشی است که رابطه‌ای بین:  
یک (زمینه خاص Context)، یک (مسأله Problem) و یک (راه حل Solution) را  
بیان می‌کند.

❖ الگو به عنوان یک (مؤلفه Element)  
در دنیای واقعی، هر الگو یک رابطه بین یک زمینه خاص، یک سیستم خاصی  
از نیروها که مکرر در این زمینه اتفاق می‌افتند و یک پیکربندی سه بعدی  
خاص که اجازه می‌دهد این نیروها خودشان را تجزیه نمایند

❖ الگو به عنوان مؤلفه زبان، یک (دستورالعمل Instruction) است  
که نشان می‌دهد این پیکربندی سه بعدی می‌تواند بارها و بارها دوباره  
استفاده شود تا بتواند هر جا که زمینه آن را مناسب بداند، سیستم نیروها را

❖ الگو یک (شیئی Thing) تجزیه‌کننده است  
که در دنیای واقعی رخ می‌دهد، و یک قاعده است که به ما می‌گوید که چگونه  
و چه زمانی باید آن را بسازیم

❖ الگو به عنوان یک (فرایند Process) و هم یک شیء:  
توصیفی از یک فرایندی است که آن شیئی را تولید خواهد کرد

# ویژگی الگوهای معماری نرم افزار

- يك الگو مسأله طراحی (دوباره ظاهرشوندهاي (Recurring design problem) را مشخص مي کند
- که در (موقعیت طراحی مشخصي (Specific design situation) رخ دهد، ويك راه الگوها (تجارب طراحی خوب (اثبات شده و موجود (Solution) (Present) براي آن (نشان مي دهد) Existing , well-proven (design experience) را (مستند Document) مي کند.
- يك الگو (تجريد هايي (Abstraction) را (شناسايي Identify) و (مشخص Specify) مي سازند
- که اين تجريد ها بالاي سطح (کلاس هاي منفرد (Single classes) و (نمونه ها Instances) و يا
- يك الگو چندين کامپوننت ها و کلاس ها را فراگير مي کند و مسئوليت ها و روابط بين آنها و نیز نحوه همکاري آنها را با جزئیات بيان مي کند
- يك الگو (واژگان Vocabulary) و فهم عمومي براي (اصول طراحی Design principle) تهيه مي کند

• الگوها به شما کمک می‌کنند که معماری نرم‌افزار پیچیده و ناهمگن ایجاد نمایید

• الگوها ساختن نرم‌افزار با ویژگی‌های تعریف شده را پشتیبانی می‌کنند

• الگوها کمک می‌کنند که بتوانید پیچیدگی نرم‌افزار را مدیریت نمایید

• الگوها وسیله‌ای برای مستند نمودن معماری نرم‌افزار می‌باشند

**درهنگام استفاده از یک الگو برای حلّ یک مسأله باید آن را با ویژگی‌های خاصی استفاده نمود**



● سبک‌های معماری نیز به همان روش الگوها به منظور حل برخی مسائل و ویژگی‌های کیفی تعریف می‌شوند

● سبک‌ها راه حل‌ها نمی‌باشند بلکه چارچوبی برای راه حل‌ها می‌باشند

● سبک چیدمان و فضای مسأله را مشخص می‌نماید و الگو نحوه ارتباطات و الگوریتم‌ها را مشخص می‌سازد

● انتخاب یک سبک محدود به حوزه فضای راه حل می‌باشد، بنابراین پیچیدگی فرایند شکل مناسب را کاهش می‌دهد

● انتخاب الگو محدود به حوزه فضای راه حل نمی‌باشد، لذا پیچیدگی فرایند شکل مناسب کاهش پیدا نمی‌کند

**بنابراین نمی‌توان گفت که الگو و سبک**

**یکی می‌باشند**

# آشنایی مختصر با (متدولوژی‌های چابک Agile Methodologies)

متدولوژی‌های چابک در اواسط دهه نود معرفی شدند و عمدتاً بر پایه اصول و قواعدی برای طراحی، برنامه‌سازی، و تست می‌باشند که منجر به افزایش انعطاف پذیری و کاربری فرایند توسعه

سیستم‌های نرم‌افزاری می‌شوند. اکثراً متدولوژی‌های چابک دارای فرایندی چابک متعددند، و در این‌جا نام هفت متدولوژی‌های چابک معروف را ذکر می‌کنیم

- 1.

متدولوژی DSDM

متدولوژی Scrum

متدولوژی XP

متدولوژی ASD

متدولوژی dX کریستال

شفاف‌سازی متدولوژی Crystal Feature Driven Development

FDD

خلاصه مقایسه میان پشتیبانی و یا عدم پشتیبانی از فعالیت‌های موجود در فازهای مختلف 7 متدولوژی فرایند چابک (فرایند عمومی SDLC=System Development Life Cycle)

## خلاصه‌سازی و مقایسه فعالیت‌های موجود فرآیندها، فاز پیش از پروژه

متدولوژی	امکان سنجی پروژه	تعامل شدید با کاربر	فراهم کردن منابع مورد نیاز پروژه	استخراج نیازمندی‌های سطح بالای سیستم	استخراج نیازمندی‌های غیر کارکردی	اولویت بندی نیازمندی‌ها	تعیین محدوده سیستم	تشکیل تیم توسعه	مدل سازی معماری سیستم	برنامه ریزی سطح بالا	فیلتر تناسب	قطعی کردن معیارهای موفقیت
SDLC	+	+	+	+	-	-	+	+	+	!	!	!
DSDM	+	+	+	+	+	+	+	+	+	+	-	-
Scrum	+	+	+	+	+	+	+	+	+	+	+	-
XP	+	+	+	+	-	-	+	+	+	-	-	-
ASD	+	+	+	+	-	+	+	+	+	+	-	+
dX	--	+	-	+	-	-	+	+	+	+	-	-
Crystal Clear	+	+	+	+	-	+	+	+	+	+	-	-
FDD	--	+	--	+	+	+	+	+	-	+	-	-

خلاصه مقایسه میان پشتیبانی و یا عدم پشتیبانی از فعالیتهای موجود در فازهای مختلف 7 متدولوژی فرایند چابک (فرایند عمومی SDLC=System Development Life Cycle)

## خلاصه سازی و مقایسه فعالیتهای موجود فرآیندها، فاز پروژه

متدولوژی	برنامه ریزی تکرار	تعامل شدید با کاربر	تحلیل و طراحی در صورت لزوم	پیاده سازی	انجام تست به صورت پیوسته	ارزیابی منظم و پیوسته	تجمع تدریجی نرم افزار	انتقال تدریجی سیستم به محیط کاربری	مستند سازی	مرور تکرار
SDLC	-	+	+	+	+	+	+	-	+	!
DSDM	+	+	+	+	+	+	+	+	+	+
Scrum	+	+	+	+	+	+	-	-	+	+
XP	+	+	+	+	+	+	+	+	+	+
ASD	+	+	+	+	+	+	+	-	+	+
dX	+	+	+	+	+	+	+	+	+	+
Crystal Clear	+	+	+	+	+	+	+	-	+	+
FDD	+	+	+	+	+	+	+	-	-	-



خلاصه مقایسه میان پشتیبانی و یا عدم پشتیبانی از فعالیتهای موجود در فازهای مختلف 7 متدولوژی فرایند چابک (فرایند عمومی SDLC=System Development Life Cycle)

## خلاصه سازی و مقایسه فعالیتهای موجود فازهای فاز ساز بر رفته

متدولوژی	راه اندازی سیستم در محیط کاربری	تست مقبولیت	پشتیبانی	تکمیل مستندات	آموزش کاربران	اعلام مرگ محصول	فعالیت های پس از مرگ
SDLC	+	+	+	+	+	+	!
DSDM	+	+	+	+	+	--	--
Scrum	+	+	+	+	+	--	--
XP	+	+	+	+	+	+	+
ASD	+	+	+	+	+	--	+
dX	+	+	+	+	+	--	--
Crystal Clear	+	+	+	+	+	--	+
FDD	--	--	--	--	--	--	--

# ضعف‌های موجود در فرایند عمومی SDLC در تقابل با دیگر فرایندهای چابک

❖ عدم انجام اولویت‌بندی بر روی نیازمندی‌های سطح بالا در فاز آغاز

❖ عدم وجود برنامه‌ریزی سطح بالا در فاز آغاز

❖ عدم وجود انعطاف‌پذیری و قابلیت تنظیم پویا

❖ عدم راه اندازی محصول به صورت تدریجی در هرتکرار

❖ عدم وجود فعالیت مرور تکرار

❖ فعالیت پس از

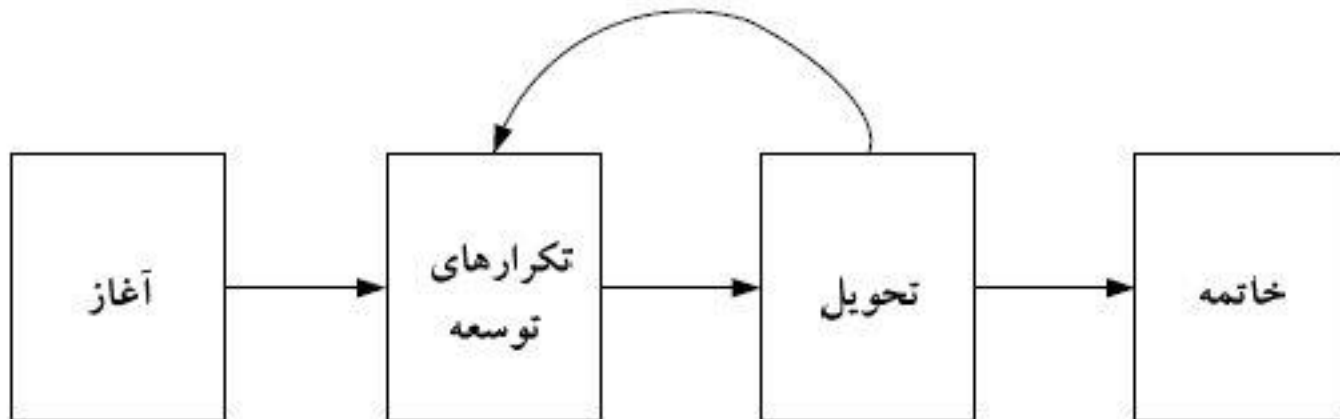
مرگ

# ارائه فرایند چابک عمومی بهبود یافته

فرایند چابک عمومی بهبود یافته  
با چهار فاز:

◆ آغاز ◆ تکرارهای توسعه  
◆ انتشار ◆ مرگ

فاز تکرارهای توسعه، چرخه اصلی تولید این فرایند بوده و از چند تکرار تشکیل می‌شود.



فرایند چابک عمومی  
بهبود یافته



تنها کتاب ارائه شده تاکنون در زمینه الگوهای فرایند شیئگرا  
توسط Scott W. Ambler  
ارائه شده است و الگوی فرایند را بدین شکل تعریف کرده است:

**“ الگوئی است که ره یافت یا مجموعه ای از عملیات  
اثبات شده ی موفق برای  
توسعه نرم افزار را تعریف می کند ”**

# طبقه‌بندی فرایندی امبلر برای الگوهای Process Patterns

امبلر الگوهای فرایند را به سه  
دسته تقسیم کرد

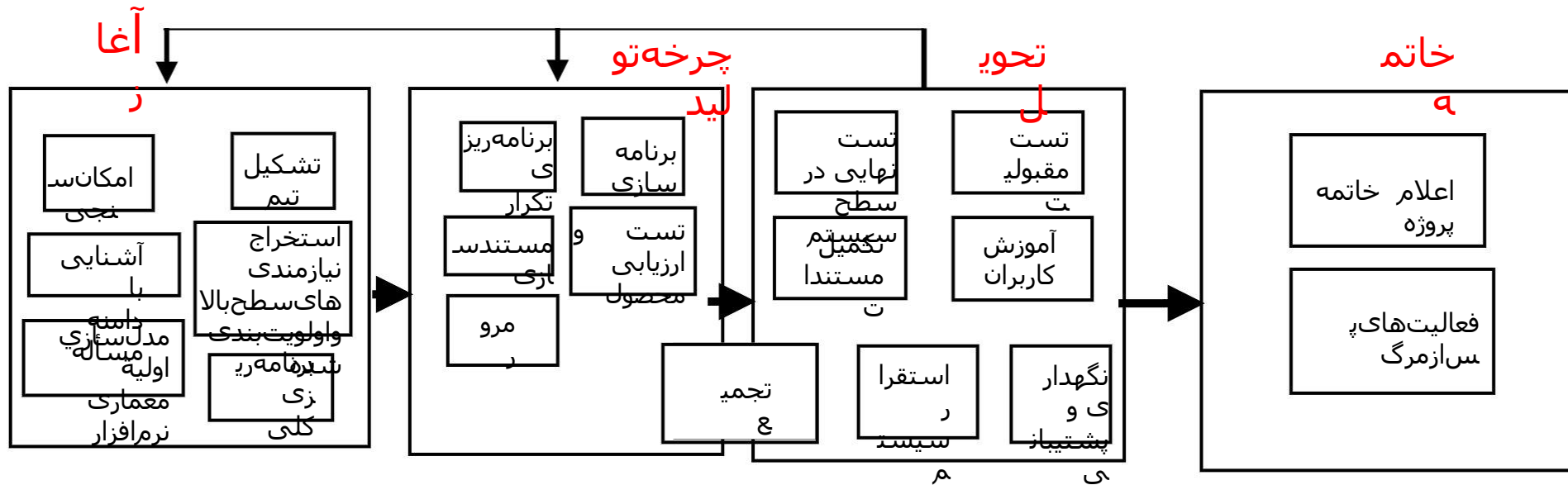
◆ الگوی فرایند وظیفه Task Process  
Pattern

◆ الگوی فرایند مرحله Stage Process  
Pattern

◆ الگوی فرایند فاز Phase Process Pattern

در هر فرایند فازها به صورت ترتیبی و مراحل درونشان به صورت  
تکراری اجرا می‌شوند

# الگوهای فرایند فاز Phase Process Pattern




## متدولوژی الگوهای فرایند فاز Process Phase Pattern



در این ارائه ابتدا تعریف مفاهیم پایه‌ای، وبعد از آن هفت متدولوژی چابک بررسی شد و در ادامه

تقابل میان مراحل موجود در دوره عمر هر یک از آنها و فرایند عمومی ارائه شده توسط امبلر مورد

بررسی قرار گرفت. سپس با توجه به نتایج بررسی‌ها و کمبودهای فرایند عمومی این فرایند را با اعمال حداقل تجرید، توسعه داده و فرایند چابک عمومی بهبود یافته ارائه گردید. در بخش پایانی، بر مبنای طبقه‌بندی ارائه شده توسط امبلر در زمینه الگوهای فرایند شیئگرا، الگوهای فرایند چابک در سطح فاز و مرحله مورد بررسی قرار گرفت.



با تشکر از  
توجه شما  
نو پروژه

*2pro.ir*