

فصل سوم – توسعه چالاک

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e*, by Roger S. Pressman

Presented by: Hassan Tanabi

Hassan.Tanabi@Gmail.com

فهرست مطالب

- چالاکي چیست؟
- مدل های فرایند چالاک
 - مدل XP
 - سایر مدل ها



ایده اصلی

- در مدل های فرایند ذکر شده در فصل قبل، خطاهای انسانی نادیده گرفته شده است. با وجود این خطاها نظم زیاد منجر به شکست می گردد.

- ”ما راه های بهتری را برای توسعه نرم افزار با انجام آن و کمک به دیگران برای انجام آن، نشان خواهیم داد“

- در طی این کار ما به ارزش های زیر دست یافته ایم :

- اهمیت اشخاص و ارتباطات نسبت به فرایند و ابزار

- اهمیت نرم افزار عملیاتی نسبت به مستندات کامل

- اهمیت همکاری مشتری نسبت به مذاکرات در حین قرارداد

- اهمیت پاسخ به تغییرات نسبت به پیگیری یک طرح اولیه

توسعه چالاک در یک نگاه

• اهمیت:

- سیستم های نرم افزاری جدید دائم در حال تغییر هستند

• هدف:

- رضایت مشتری، تحویل زودهنگام و افزایشی

• توسط:

- مهندسين نرم افزار و ساير ذينفعان در غالب يك تيم

• محصول کار:

- از دیدگاه مهندسين نرم افزار و مشتری یکسان است. محصول مهم و واقعی کار، یک افزایش اجرایی نرم افزار

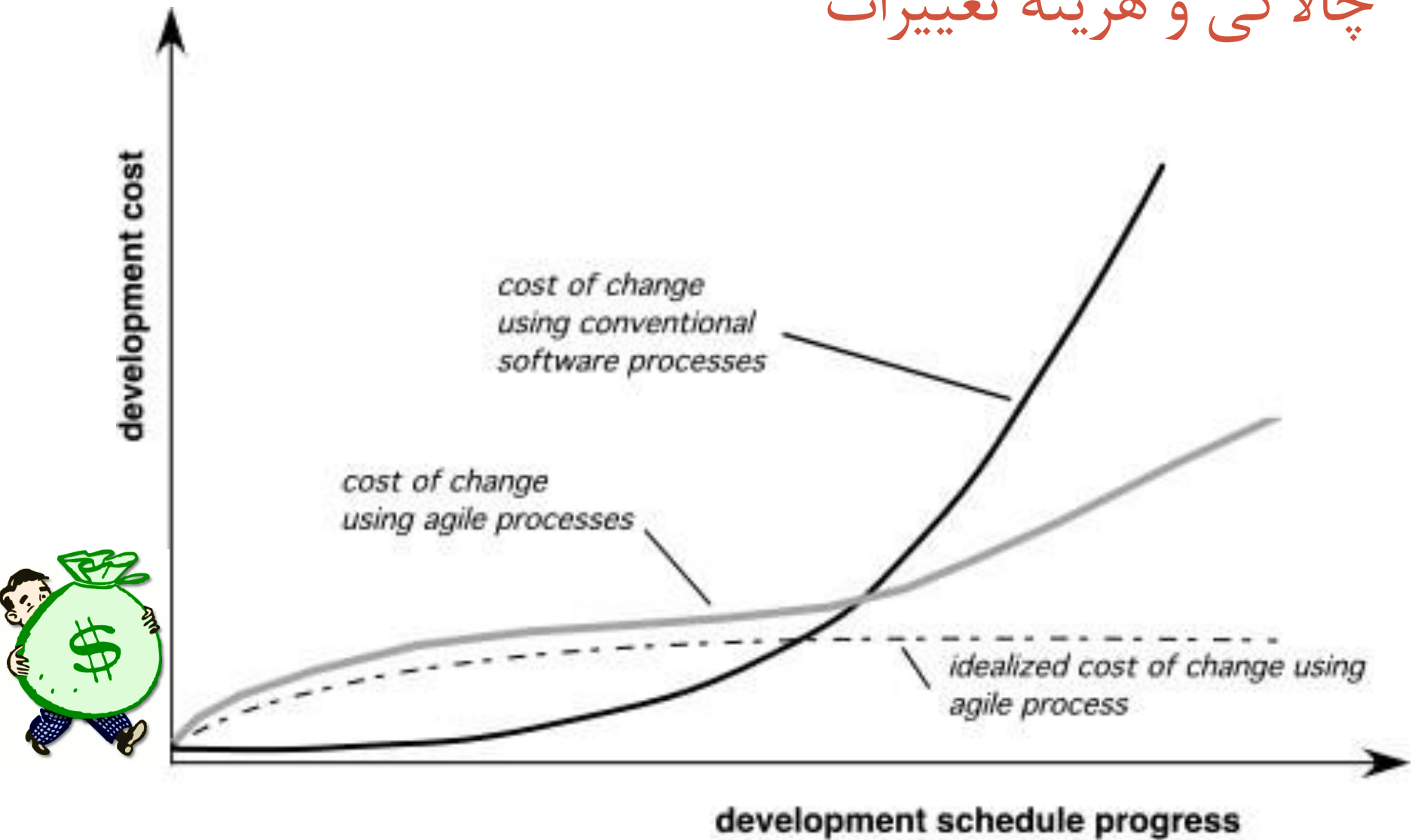
چالاکي چیست؟

- پاسخ مؤثر (سريع و مناسب) به تغييرات
- برقراری ارتباط مؤثر میان تمامی ذینفعان
- دیدن مشتری به عنوان عضوی از تیم توسعه (حذف “ما”، “آنها”)
- برنامه ریزی در یک دنیای غیرقطعی محدودیت هایی دارد، بنابراین برنامه ی پروژه باید انعطاف پذیر باشد.
- تشکیل یک تیم برای اینکه کار انجام شده تحت کنترل باشد.

• در نتیجه:

تحويل سریع و افزایشی نرم افزار به کاربر

چالاکي و هزينه تغييرات



فرایند چالاک

- از نیازهای ارائه شده (سناریوها) توسط مشتری نشأت می‌گیرد.
- طرح‌ها، کوتاه مدت هستند.
- توسعه نرم‌افزار به صورت چرخشی با تاکید زیاد بر روی فعالیتهای ساخت سیستم
- تحویل نرم افزار در چندین افزایش
- در صورت تغییر، با تغییرات تطابق می‌یابد.



اصول چالاکی

- (1) اولویت اول، راضی کردن مشتری با ارائه سریع و پیوسته ای از نرم افزار
- (2) استقبال از تغییرات، حتی در مراحل انتهایی توسعه
- (3) ارائه متناوب نرم افزار، از چند هفته تا چند ماه با اولویت کوتاه تر بودن این زمان
- (4) ذینفعان و تیم توسعه باید با یکدیگر در طول پروژه همکاری (روزانه) کنند.
- (5) ساختن پروژه حول افراد، دادن فضا و امکانات مورد نیازشان و اطمینان به آنها در انجام وظایفشان
- (6) مؤثرترین روش برای بدست آوردن اطلاعات، گفتگوی چهره به چهره می باشد.



اصول چالاکی (ادامه)

- (7) نرم افزار اجرایی اصلی ترین معیار پیشرفت است.
- (8) فرایندهای چالاک توسعه صبورانه را ترویج می دهند سرمایه گذاران، توسعه دهندگان و مشتری باید گام های نامحدودی را استوار و پایدار باقی بمانند.
- (9) توجه پیوسته به کیفیت فنی و طراحی چالاک
- (10) سادگی: به همان اندازه که نیاز داریم کار انجام دهیم.
- (11) بهترین معماری ها، نیازمندیها و طراحی ها از تیم های خود سازماندهی شده پدیدار می شوند.
- (12) در هر دوره زمانی تیم متوجه می شود که چگونه کارا تر عمل می کند، سپس رفتار خود را براساس آن تنظیم می نماید.

فاکتورهای انسانی

- ویژگیهای مهمی که در افراد تیم چالاک و خود تیم باید وجود داشته باشد:
- **شایستگی:** هوش، توانایی مرتبط، دانش عمومی در مورد فرایند انتخابی
- **تمرکز عمومی:** علاوه بر دانش اختصاصی، تمرکز بر روی یک هدف مشترک، ارائه افزایش قابل اجرا از نرم افزار در زمان معین
- **همکاری:** علاوه بر همکاری در فعالیت های فرایند (آنالیز، طراحی، ...)، همکاری در جمع آوری اطلاعات برای ذینفعان در مورد فعالیت تیم، ارزش کار برای مشتری، ...
- **توانایی تصمیم گیری:** استقلال داخلی تیم، توانایی تصمیم گیری تیم در موارد فنی و یا تصمیمات مرتبط با پروژه

فاکتورهای انسانی (ادامه)

- توانایی حل مسئله فازی: مشکلی که امروز حل می کنیم لزوما فردا وجود ندارد، اما حل این مشکل (حتی مشکل اشتباه) ممکن است بعدا برای ما مفید باشد.

- اعتماد و احترام متقابل: یکپارچگی بهتر از تکه تکه بودن است.

- خود-سازماندهی:

1. تیم خودش را برای انجام کار سازماندهی کند.
2. تیم فرایند را به منظور داشتن بهترین محیط سازماندهی کند.
3. تیم برنامه کار را به منظور ارائه بهترین افزایش نرم افزار سازماندهی کند.

مدل XP

- فرایند چالاکي که بیشترین استفاده را دارد. ارائه شده توسط آقای Kent Beck
- XP از یک رویکرد شی گرا استفاده می کند.
- اجرای مجموعه ای از قوانین و تمرینات، که تحت چهارچوب چهار **فعالیت** انجام می شود:

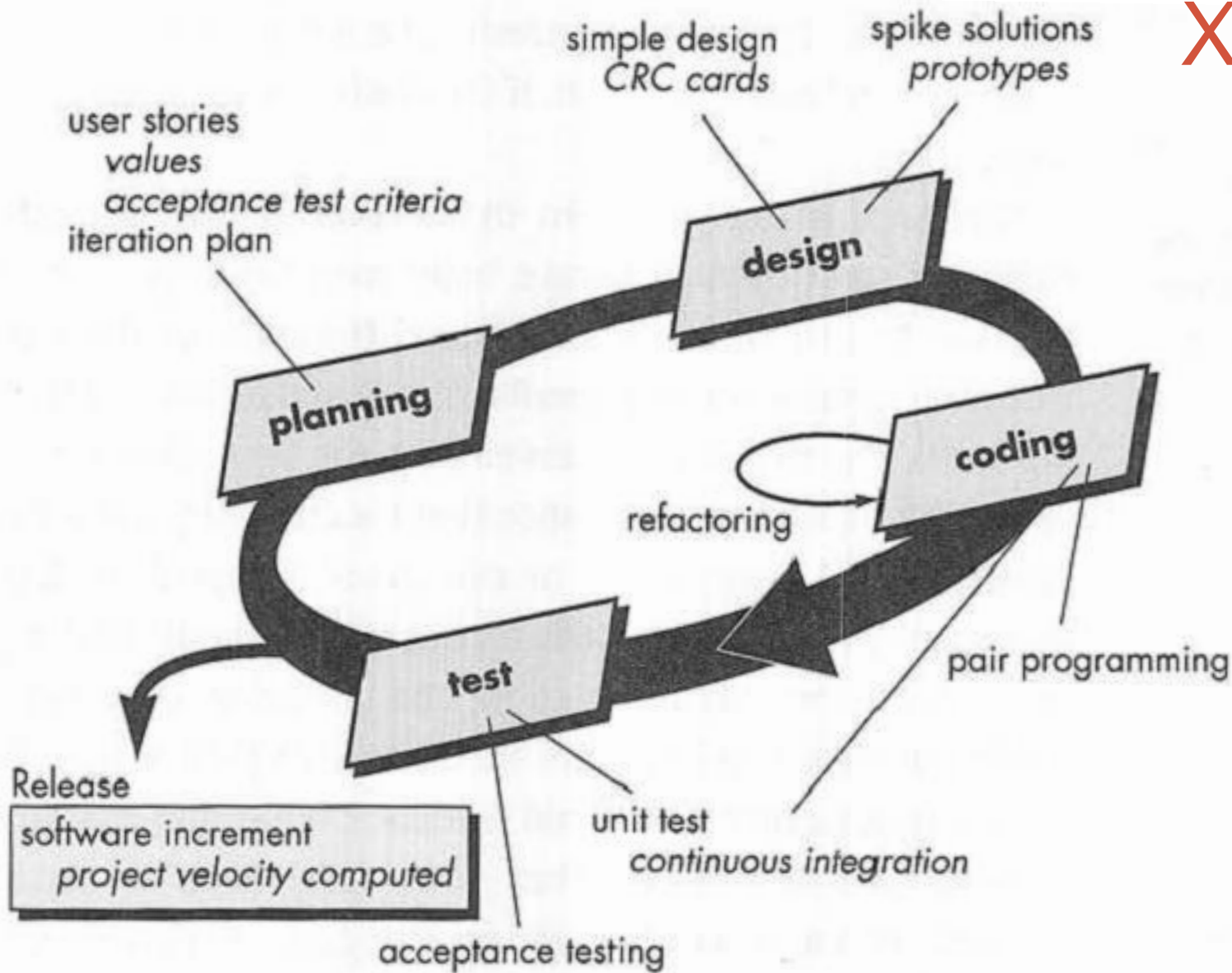
1. برنامه ریزی Planning

2. طراحی Design

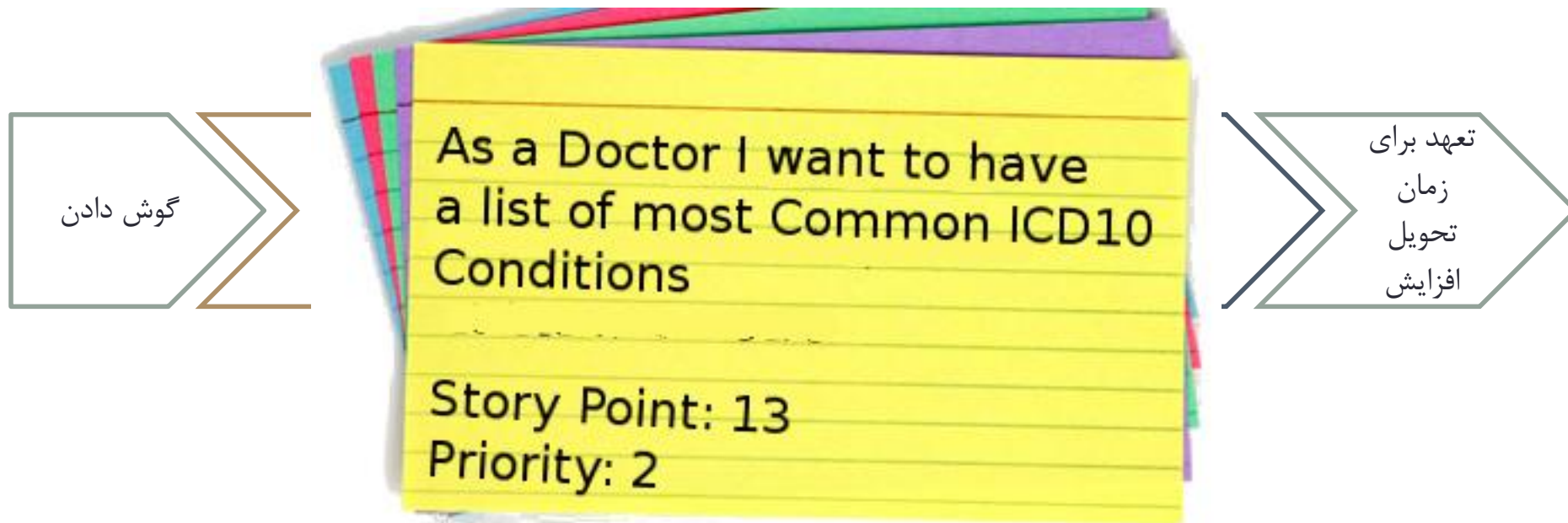
3. کدنویسی Coding

4. تست کردن Testing

مدل XP



مدل XP - برنامه ریزی



- هر داستان کاربری توسط مشتری بر روی یک کارت نوشته می شود، سپس مشتری به هر کدام ارزش (اولویت) می دهد. اعضای تیم هر داستان را بررسی و هزینه آنرا تخمین می زنند (چند هفته برای توسعه نیاز دارد).

مدل XP - برنامه ریزی (ادامه)

- بعد از ارائه اولین انتشار ارزیابی سرعت پروژه

تعداد داستان های کاربری
پیاده سازی شده طی
اولین انتشار

• اهداف:

- تخمین تاریخ های تحویل
- وجود نداشتن سربار داستان کاربری در یک افزایش
 - تغییر محتوی در انتشار
 - تغییر تاریخ تحویل

مدل XP - طراحی

- طراحی در XP به شدت از قاعده KIS پیروی می کند.

Keep It
Simple

- استفاده از کارت های CRC

Class
Responsibility
Collaboration

- سازماندهی کلاس های شی گرا که در این افزایش دخیل هستند.

مدل XP - طراحی (ادامه)

• نمونه از کارت های CRC

Class: FloorPlan	
Description	
Responsibility:	Collaborator:
Defines floor plan name/type	
Manages floor plan positioning	
Scales floor plan for display	
Scales floor plan for display	
Incorporates walls, doors, and windows	Wall
Shows position of video cameras	Camera

مدل XP - طراحی (ادامه)

- برای طراحی های پیچیده تر، ساختن "Spike Solution" پیشنهاد می گردد.

A Design
Prototype

- استفاده از دوباره سازی Refactoring

یکی از تکنیک های بهینه سازی طراحی. تغییر در رفتار بیرونی کد دیده نمی شود. ساختار داخلی بهبود پیدا می کند.

مدل XP - کدنویسی

- ابتدا سراغ کدنویسی نمی رویم.
- توسعه سری تست واحدها برای آزمایش هر داستان کاربری، سپس توسعه کدی که این تست ها را با موفقیت بگذراند.
- هیچ چیز اضافه ای نیاز نمی باشد (KIS)

حل کردن مشکل در همان لحظه

کیفیت بالاتر: کد در همان لحظه بازبینی می شود

• استفاده از مزایای برنامه نویسی دو نفره

Pair
Programming

مدل XP - تست کردن

- تمام واحدها هر روز تست می شوند.
- تست واحدها در کنار یکدیگر
- آزمایش پذیرش که توسط مشتری و از داستان های کاربری گرفته می شود تا ارضا شدن وظیفه مندی های مورد نظر مشتری اطمینان حاصل گردد.

توسعه نرم افزار توافقی ASD

این مدل توسط آقای Jim Highsmith ارائه گردید.

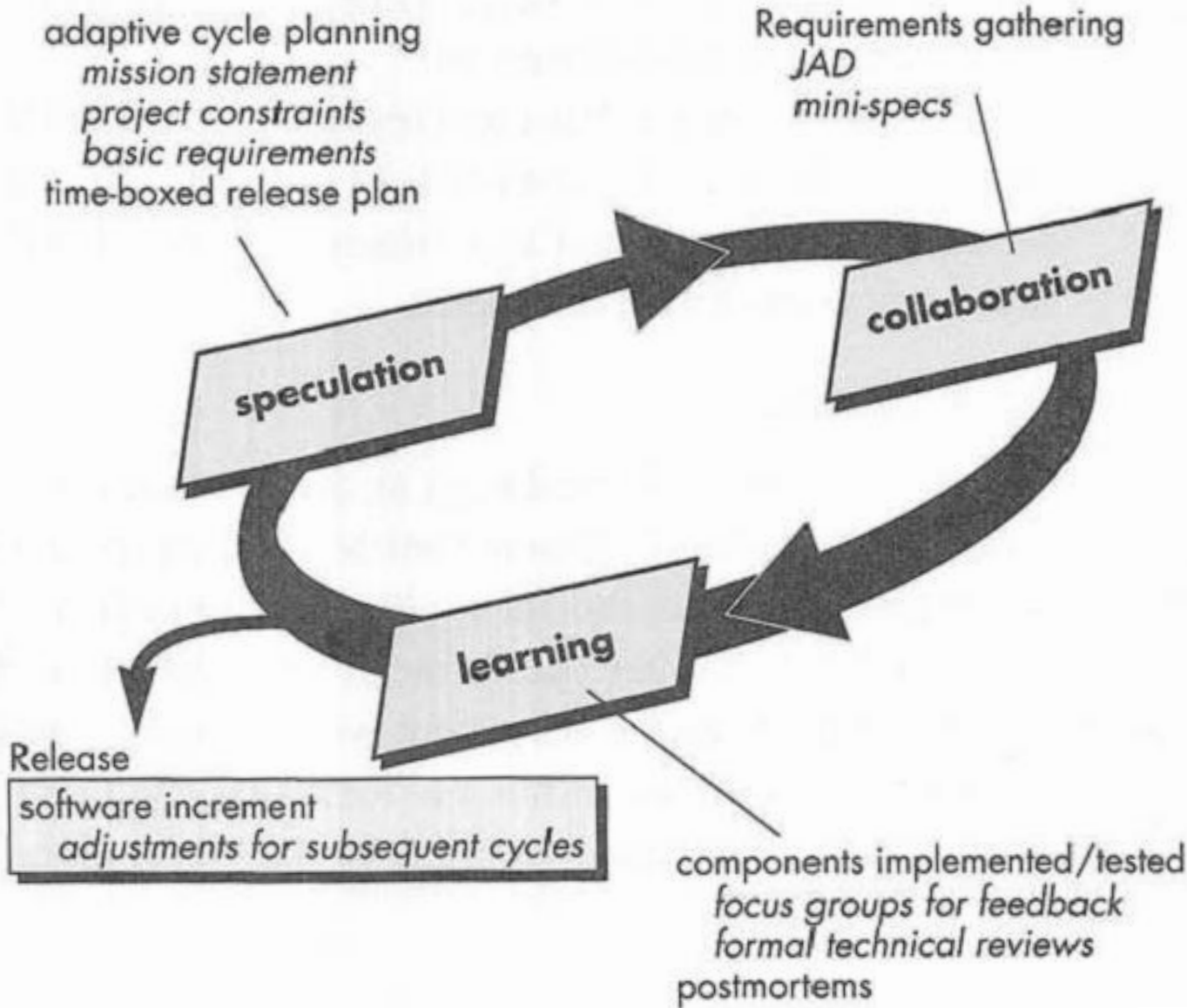
این فرایند از سه فعالیت اصلی تشکیل شده است:

تفکر Speculation

همکاری Collaboration

یادگیری Learning

مدل ASD



توسعه نرم افزار توافقی ASD

• تفکر:

- مشخص کردن اهداف مشتری
- محدودیت های پروژه
- تاریخ های تحویل
- نیازمندیهای اساسی

به منظور مشخص کردن مجموعه چرخه های انتشار که برای پروژه لازم است.

• همکاری:

- تأکید روی کار گروهی و تعامل به منظور **افزایش بازدهی**
- نیاز به وجود اعتماد متقابل به منظور:
 - انتقاد سازنده
 - کمک بدون رنجش
 - کار کردن به اندازه (بیشتر از) بقیه
 - داشتن توانایی برای همکاری در کار

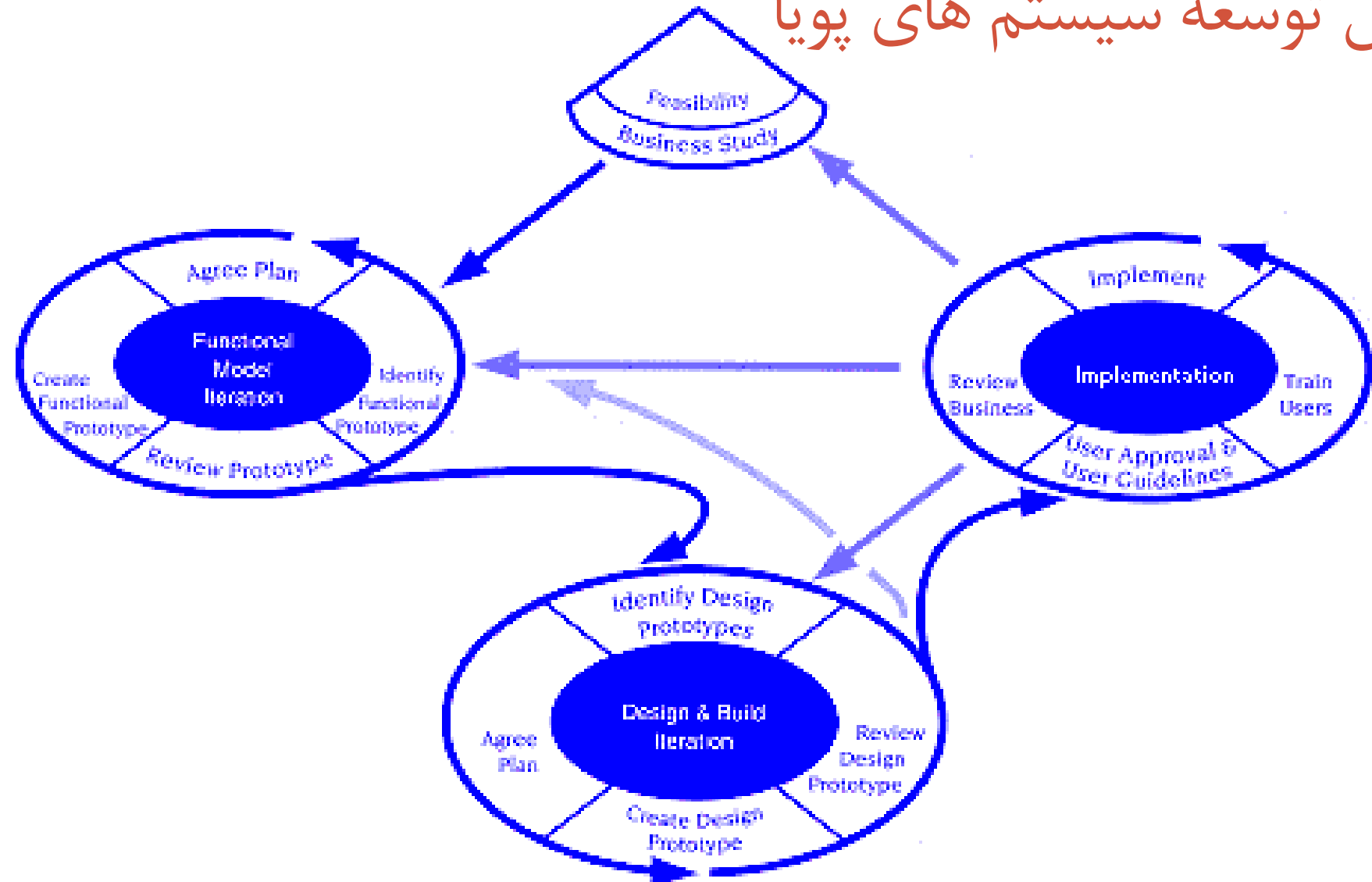
توسعه نرم افزار توافقی ASD

- یادگیری:

- با پیشروی در پروژه به سمت چرخه های پیچیده تر تأکید بر روی یادگیری می باشد.

- این یادگیری نقطه کلیدی در رسیدن به خود سازماندهی تیم می باشد.

روش توسعه سیستم های پویا



خسته نباشید

