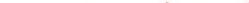


روشن‌های چاپ: {  کاربرد روش‌های چاپ: ل روش‌های بنایه‌زنی سده:

توسعه سرع نظر از از

میرزا ناصر مدنی (اورزش)

کے ازان جامیٰ نہ نہس دے،

دانلودی های سریع، حل

تغیریں سے املاں ندارد

نیازمندی پایدا رعیت کرد

رضا افرازیان و مرتضی عالمی

کارخانه‌ای نسبت به کار را بیشتر دارد.

سسته به چهار ترین های مختلف

دو سوچہ فرمیا اند

—

نادهایی از راهیت ترسیم افراد

۸۰، ۹۰ سے ایجاد رسمیتیں

)- تمرز روی کدو نه طراح

روش‌های حساب

١ - تحويل حججٍ من مصر

هدف رئیس‌های حامی:

کمیت ها

الله يحيى

نحویات تاریخی به عباره

لِلْمُؤْمِنِينَ

وَرَسُولُنَا صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ

میری جسی ری ری کاری کاری چا

- نله داری زمانی مسئله ای بر سر داشت

Curbs and steps

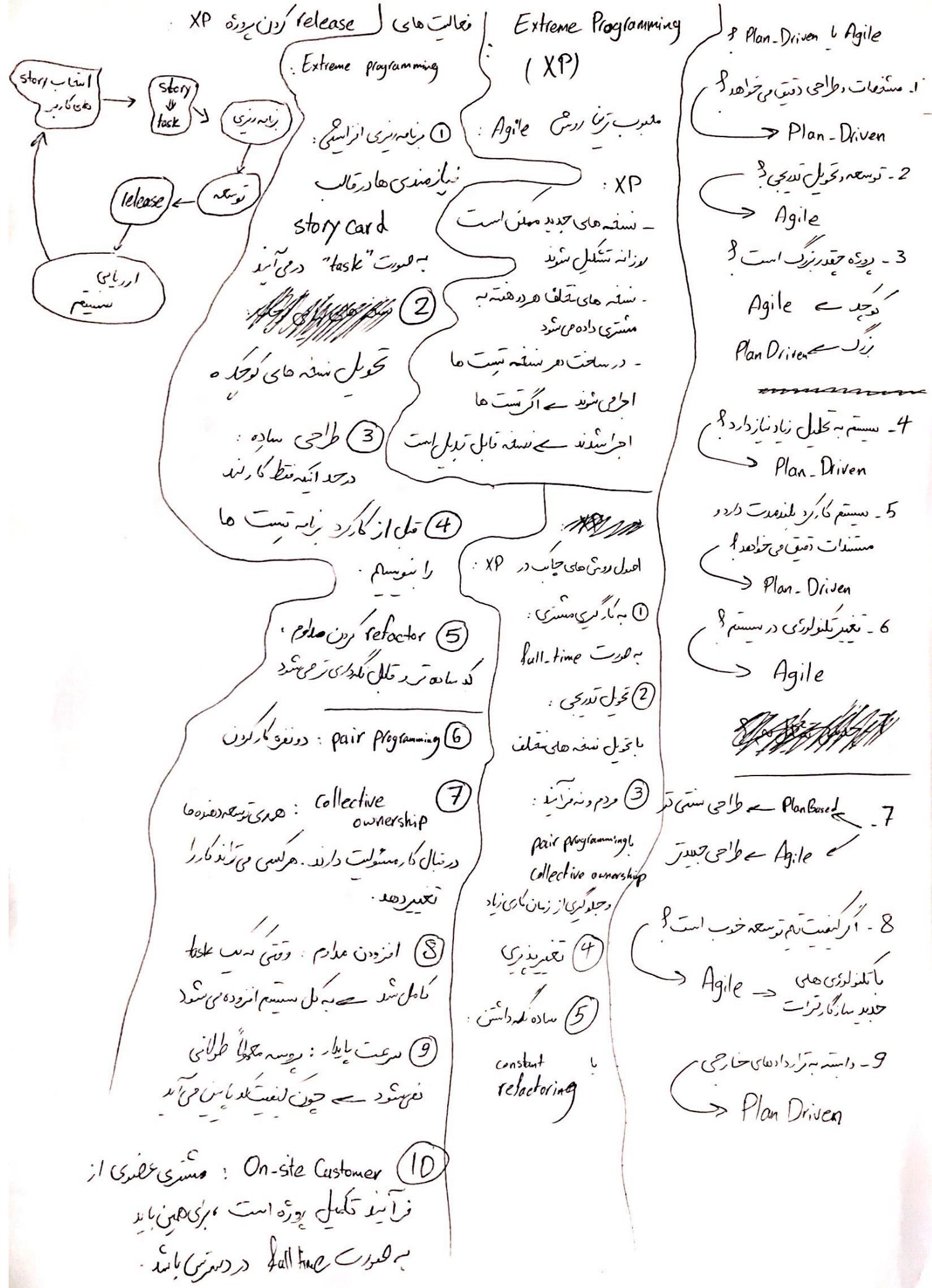
سَمِعَ بِهِ وَهُنَّ مُنْكَرٌ لِّعْنَافٍ نَّذِيرٍ يَأْمُرُ

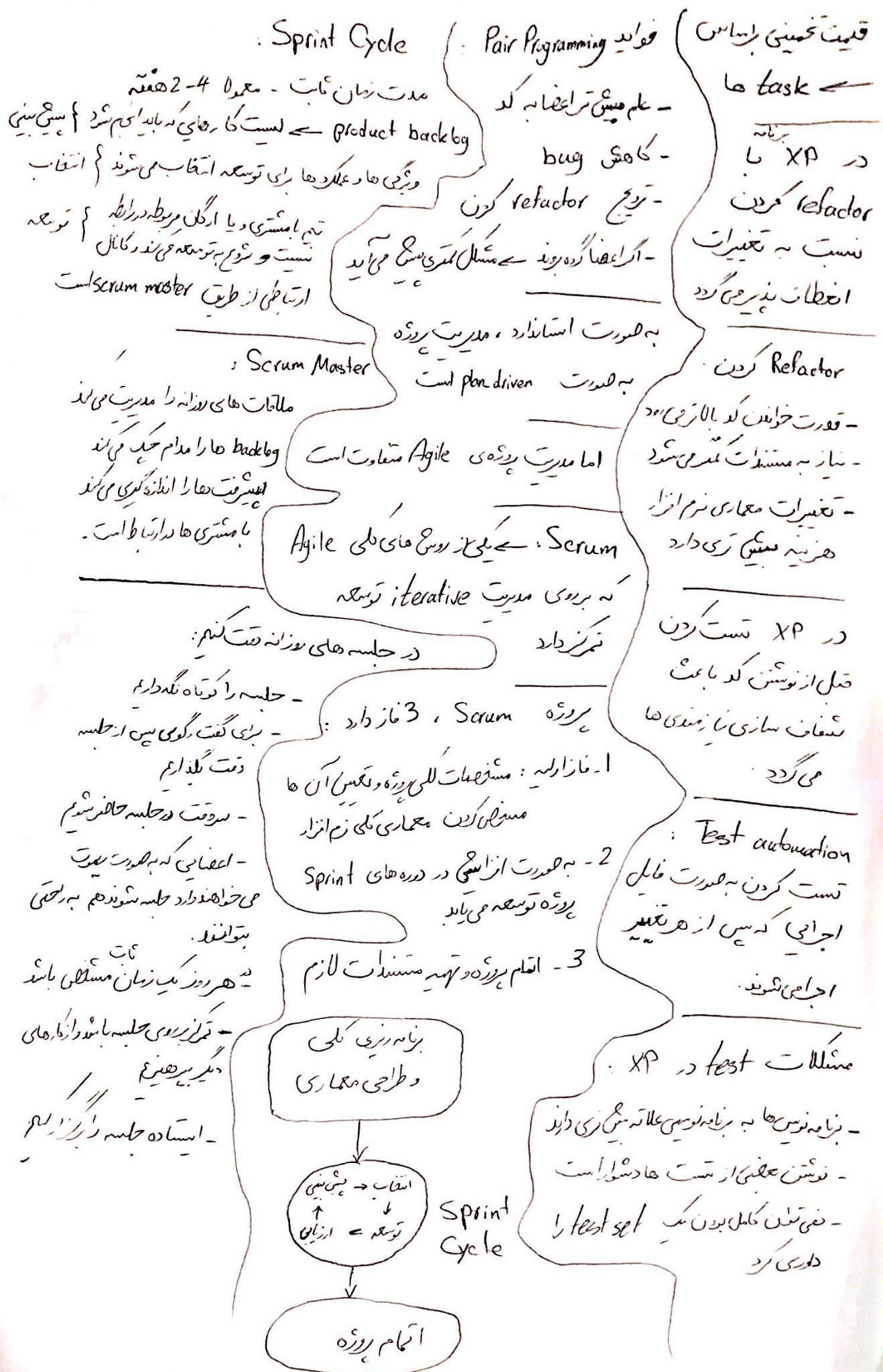
115

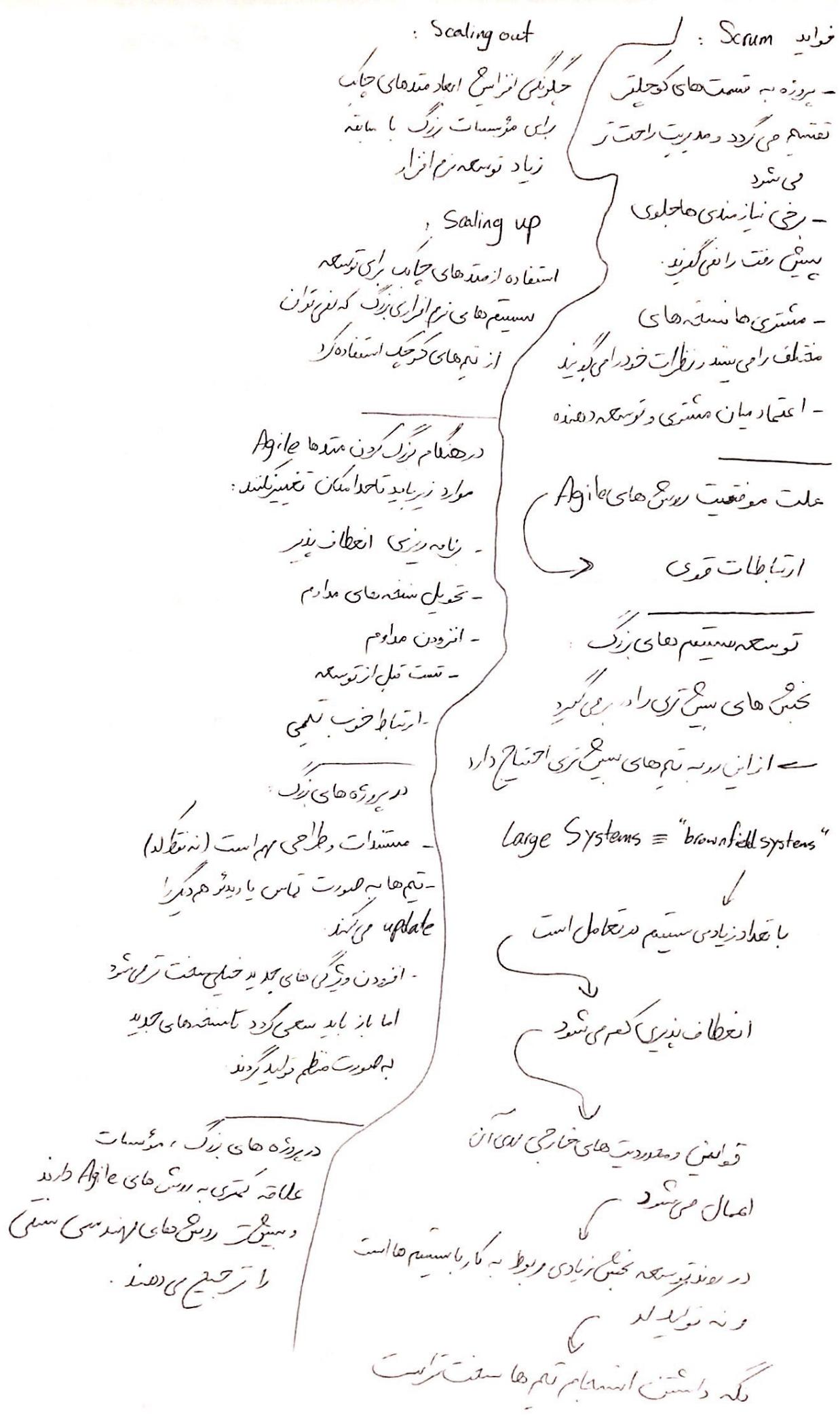
۲) ملکداری ساده :

ساده مدلن توسعه زائر را

لِمْ رَهْبَرْ سَعْدِ



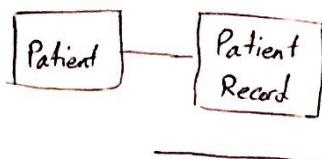






Class Diagram

درساخت System سیستم کاربری دارد
هر لاس نهایه معرفی یا چیزی در دنی واقعی است
دستور میان راهنمایی داده می شود.



Generalization

برای مریت سیستمی بروده

جای ذرازهای رای هر جن، آن ها را
لاس های مانند را بازیابی می کنند

animal, car, houses ...

و آن خیات را می دانی لاس هایی هم

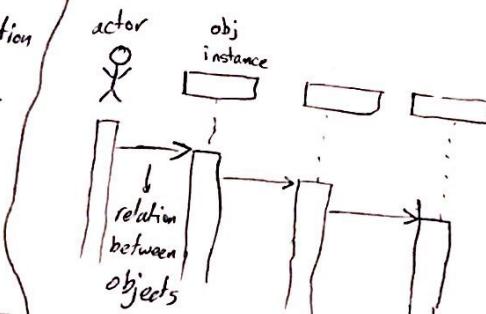
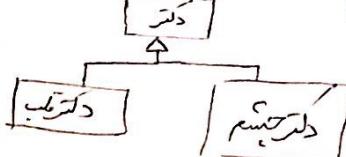
ایسا خیا را می داند در ذرازی می بین

سیب و مریت هر دو سیستم

OOP Generalization در زبان های

وراثت

Generalization خود را



مشخصاتی: اجزای ساختاری و ربطی آن هایم
را نشان می کند.

مشخصاتی: *in static* و *in System* می باشد

مشخصاتی: *in dynamic* و *عمل اجزایی* می باشد.

برای بیان طراحی مهندسی از این دستگاه است.

System Subject

interaction models

برای این سیستمی کاربری

System-to-System Interactions
- این سیستم را با سیستم دیگر این سیستم

سازی داشت میان این
می تواند کاربر سیستم و قابل استفاده باشد/بردن
آن را استفاده کند.

Use case diagrams,
Sequence diagrams

Interaction models:
① (Use-case) - (sequence)
②

Process models:
(Activity)

Use case Modeling

task می باشد و usecase

امنیت از این طریق می باشد

بازیگرها اموزد

Use-case Diagram

① include ... → Scan Item

Checkout include Calculate Total and Tax

include Payment

② extend Registration

Get Help on Registr

(MDE) : Model-driven engineering

در این مسیر از این بحث برای برآوردن مسیر
بررسی کنید. سپس برای مسیر دیگر مسیر
بررسی کنید. سپس برای مسیر دیگر مسیر
بررسی کنید.

طبقه بندی این روش ممکن است این دو نوع باشند
استراتژی بردن پردازه را از این دو مسیر
با دقت بیشتر کار نمایند.

فرایند:

- دسترسی به سکوی الگوی استریچ بود
- راحت تر برای پنهان های ممکن قابل پنهان کردن

محاذب:

- ممکن است این سکوی استراتژی بردن را نتواند
- ساده سازی کرد
- توجه همچشم ها را بین پنهان های ممکن
- هزینه بینهایی دارد

X UML: نظریت
MDE: تبدیل خودکار
به دستگیری مسیر
X UML: روشی برای تبدیل مسیر

ضرورت مسیر

X UML: روشی برای مسیر
این مسیر ممکن است در UML به مدل کشیده
نمایند.

- 1- نظریه ای مسیر
- 2- درگاه آنها (Domain Models)
- 3- برخی از مدل های مسیر
- 4- احتمالی مدل (Class Models)
- 5- دفعه های خودکاری (State Models)

در درخت سیستم و انتزاعی (System and Inheritance): Aggregation models

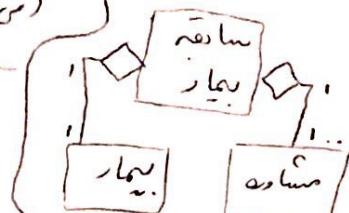
→ Data: System با دسترسی کامل است

→ Event: دسترسی محدود است (System event)

Data-Driven Modeling

سازی از System و انتزاعی
input، خروجی مناسب را در بر می کند

و در این مدل مفاهیت



Composition

سازی از System و انتزاعی

مناسب برای کلیه

نیازمندی



aggregation



Event-driven Modeling

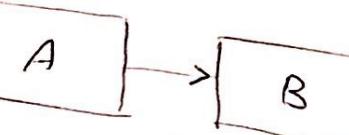
اندیل و انتزاعی

event-driven می باشد

می تواند در این طبقه قرار گیرد

سیستم می باشد

Association navigability



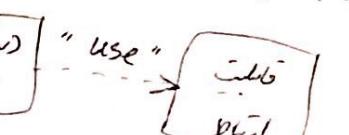
است از نوع A می تواند از این نوع دسترسی را

State Machine



State Machine

Dependency

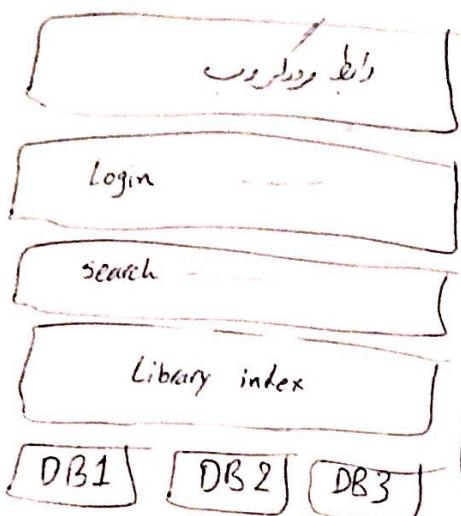


State machine

مدل های فشاری

و انتزاعی

و انتزاعی



دیدگاه های معماری: Layered Architecture

نمای معماری: Layered

نامه از طرف: to subsystem

- هر لایه یک مجموعه خدمات را تهیی می کند

- دقتی بالای تغییرات نداشته باشد

این های اطمینانی ترتیب شده اند

چه موقع استفاده می شود؟

- همچنان که بعدهم اخراج جدیدی

برای محدود سازیم که داشتیم تتم های مختلف چنین های متفاوت را

سرویس

می نیزند <-- هر کدام به سفارشی بز

امنیت حفظ لایه بود

فرایند:

- تغییری بالای راحت است

فقط یافته است از طریق تغییر در

- اجزای امن استفاده جوست افزایش

اطمینان سیستم می تواند به لایه ها

افزون شود

محاسب:

- تجزیه عیان ای ها ممکن است (سیستم اند)

و لایه های بالایی بخواهند مسکن های ای هایی باشند

در ارتباط باشند و نه از طریق لایه های میانی

- کارکرد Performance End

این هر دو خواست حفظ لایه های میانی نداشتم

هر معماری بسیار جزوی است

هر معماری بسیار جزوی است

چه زندگی: میتوان

System point -

- فرآیندهای Runtime، داده

محاسب از طریق این فرآیند

- حلقه ای خود را برپا کرده شد

4+1 نمای

logical view - 1

که class ها و این های مانند

process view - 2

Runtime to

development view - 3

برای ترسیمه کاربردی اجرا کوچکتر نمایند

physical view - 4

و آنکه خوب است اخراج در پیوندهای پایانی می شود

to scenario to use case از این است: (+1) - 5

Architectural patterns

که اینها برای زیستی، به استقلال نهادن

و استفاده محدود از داشت.

که درین هایی نسبت به دیگر طرحی های

که اینها کاری در راه رفته اند بکار رفته اند یا نه اند

خوب یا هر چهار دنیا

که درین جایی از اینها داشتند

که درین جایی از اینها داشتند