DDD ①

design patterns هو عبارة عن مجموعة من الـ
موجودة فعلياً مع إضافة بسيطة
عشان أعرف أركز على الـ bussins
عبارة عن حاجتين

① stratigic patterns (tools)
② Tactikal paterns (Tools) ⇓ الحاجات اللي ليها علاقة بالكود

① stratigic →
هفكر في الـ stratigy الـ ازاي من ناحية الـ bussenis
معنى بـ إي بقا؟!

① الـ bussines الـ اللي عايز اعمله system
وأحاول أوصل لكل تفاصيله
② ابدأ أكسره لأجزاء صغيرة (subdomains)
طبعاً أكون عارف العلاقة بينهم ايه
③ تعرف بقا مصطلحات كل subdomain
④ ابدا بقا أفكر في solution للموضوع أو الـ bussins بتاعي

~~② Tactical pattern~~

يعني إي domain ⇐ يعني من الآخر أنا بحل مشكلة
أي؟! مثلاً بحل مشكلة الطيار
مثلاً amazon أو كذا ...

يعني أنا عندي Big Domain ولتكن amazon
هبدأ أقسمه لـ subdomains صغيرة وابدأ اعرف
كل subdomain اي لزمته ومكانه في المشروع

هل هو
① core subdomain
② Geniric "
③ supporTing "

① Core subdomain ⇒ الحاجة الأساسية للمشروع
اللي بتميز المشروع بتاعي عن غيري وبتكون أكترهم تعقيد
وأكترهم تعديل و developing
Competive edge ⇒ دي الحاجة الفارقة والمميزة ليا عن غيري

② Generic subdomain ⇒ دي مش مميزة للمشروع ②
ولا حاجة دي عادية لكنها بتكون complex
مثلاً الـ Hr عام لأي مكان عادي
ومش بجدد عليه develop باستمرار

③ supporTing subdomin ⇒ غالباً
مش معقد وداخل
كمان مش بيكون عليه dev باستمرار ولا مش مميز ليا
عن غيري زي مثلاً إن أصنف الأسعار
للمنتجات في amazon

Domain Model →
تعبير عن الـ Bussnes بتاعي أعرف أعبر عنه

~~Context Map~~

~~Bounded Context~~

TakTical — Tactical pattern → # الجزء العملي أو جزء الكود
بيكون عندي منها

- Value object - EnTiTies - services
- Domain EvenTs - modules - AggregaTs
- FacTories - RepositoRies

Domain Models
① Value object ② EnTiTy

→ very simple object has some values
كمان ملوش Identity زي مثلاً Id #

ex
Class Color { RGB هنا الـ 3 values
inT red; ⇒ 2 obj مش زي بعض في
inT green; يبقا أكيد الاتنين مش
inT Blue; } نفس اللون

Color C_1 = new Color(1,1,1); ⇒ (C_1 == C_2) → True
C_2 = " "
هنا مش هقارن بالـ ref هنا لازم أقارن بالـ values الـ وجوه

③ value object ⇝

دايما فيه الـ Validation بتاعته وعلوها ال Domain models بيكون فيها الـ Validation في نفس الـ class بتاعها

المميزات

① بعمل Encapsulate للـ business داخل الـ Class

② self validated

خواصه: ① equality لازم كل الـ value تتساوى مع بعضه ② Immutable ③ Side-effect Free

② Entity

بيكون له حاجة مميزة بقارن بيها الـ obj دا بغيره لو زي بعضه يبقا نفس الـ obj لو مختلف يبقا مختلفين زي مثلا الـ Id

```
class person {
  String NId;
  Name;
  int age; }
```

هنا بقارن الـ 2 persons ببعض ⇒ لو الـ NId زي بعضه يبقا الاتنين نفس الشخص ⇓ ex

```
person p1 = new person("01211", "ib20", 13);
—      p2 = ______   ( " , "hemada", 17);
```

هنا لان الـ Id واحد ↶

p1 == p2 ⇒ True

خواصه

① unique identifier ex ⇝ ID

② mutable ⇒ عادي ممكن أغير أي حاجة ما عدا الـ unique identifier أو عموما الـ ID

③ self validate ⇝ لنفسه ↳ جواه ⇝ his own behaviors

مثال حلو عن الفرق بين DDD ولو انا مش بستخدم الـ DDD وبعمل CRUD ↑

لو بعمل CRUD عشان أعمل Terminate for employe

① هجيب الـ user

② هعمل check إنه هيتعمل له Terminate

③ هروح الـ Account بتاعه وأشيل أي Access ليه

④ هروح أعمل ايقاف لأي مرتبات بياخدها.

لكن في الـ DDD

هعمل زرار Terminate وبعدها أعمل وأظهر له كام ديفولت سبب الـ Termination والـ system بقا هيعمل كل حاجة بدون الاعتماد على الـ user لتفادي الأخطاء

④ Agrigates ⇝

مجموعة من الـ Domain models (value obj و Entity)

له رأسهم بقا الـ Agrigate Root ← بتكون دايما الـ Entity ##

لي بنعمل كدا؟؟

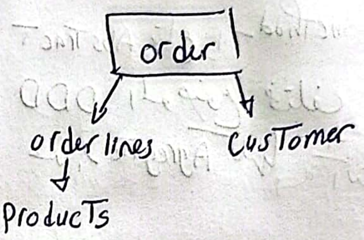
له عشان أحط الـ Business logic ← من الـ class والـ leave و complex

مثال عن Agrigates

عندي الـ order والـ order دا تحت منه orderlines تحت منه products

وكمان في customer

يبقا عندي



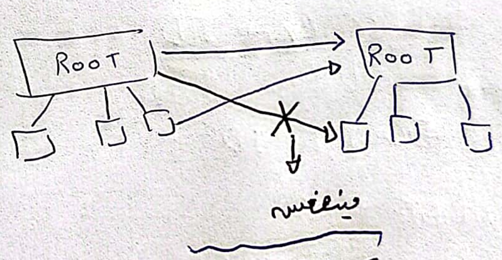
يبقا كدا عشان أوصل لأي حاجة هنا لازم تكون عن طريق الـ order

⇐ في Rules عشان يحصل تواصل بين Agregates

① الـ Roots تكلم بعضها عادي

② الـ leve ممكن تكلم الـ Root

③ الـ Root مينفعش يكلم الـ leav لازم يروح للـ root ومن الـ Root يكلم الـ leaves



Domain Events

- هو الـ Event الـ حصل في أي process
- المسئول عنها هو الـ Agregate root
- imutable ← حاجة خلاص حصلت هغير فيها ازاي!
- ليها Time stamp ← حصلت امتا؟
- ليها unique identifier (Id) عشان أعرف أفرق بين كل event وبتاني
- بعمل لها publish عن طريق الـ Agregate root

(5)

service

دى بستخدمها لو هكتب جواها حاجة عن external ال Domain أو حاجة ليها علاقة بس مينفعش اكتبها داخل ال Agregate # عندى
- أو لو ها جيب ملخص علاقة ببعض داخل method هعملها في ال service

Repository

من الطبيعي أنها تبكون لكل جدول عندي
لـ بتعمل Hide للكود عن طريق استخدام ال AbsTracT بتاع ال method وبس كدا
بس في ال DDD الموضوع مختلف
لـ مسئول عن ال Agregate كلها والى ضمنيا تبكون اكتر من جدول #