EECS3311 Software Design (Fall 2020)

Q&A - Lecture Series W6

Tuesday, October 27

concrete UI

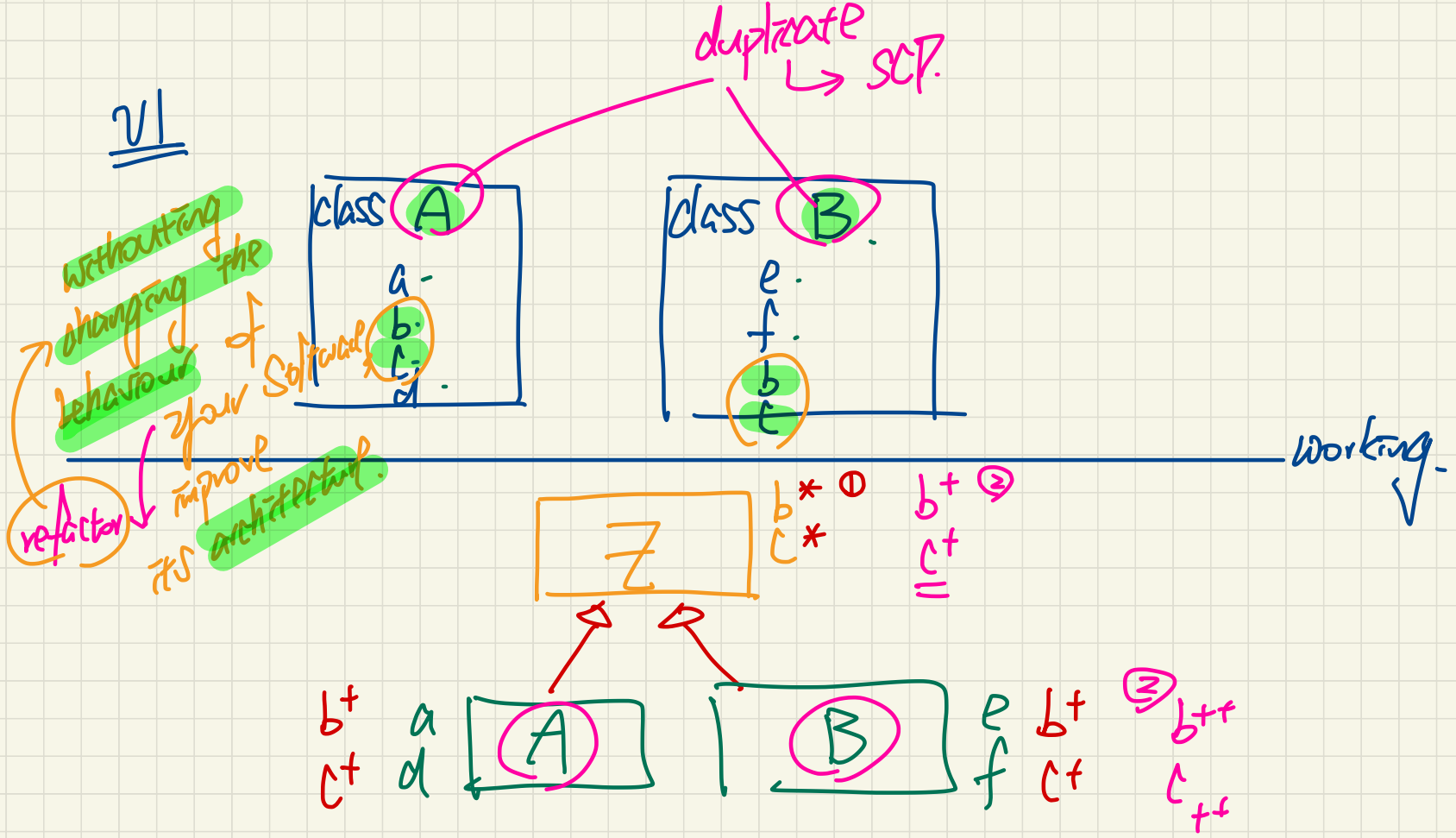
synchronous

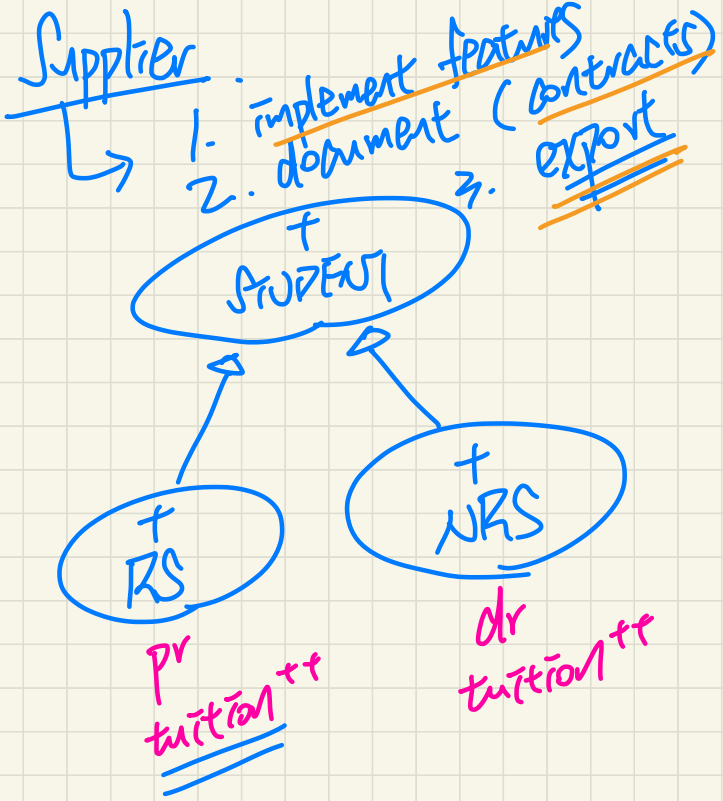
touch screen ←
insert card ←
pin ←
withdraw ←
keyboard ←
amount ←

abstracted

withdraw(
id: Int ; a: Int)

interaction with concrete UI (asynchronous)


VI
without changing the behaviour of your Software
improve its architecture
refactor
duplicate
↳ SCP.
class A
a
b
c
d
class B
e
f
b
c
working
Z
b* ①
c*
b+ ②
c+
=
b+ a
c+ d
A
B
e b+
f c+
③ b++
c++



Client

s : STUDENT

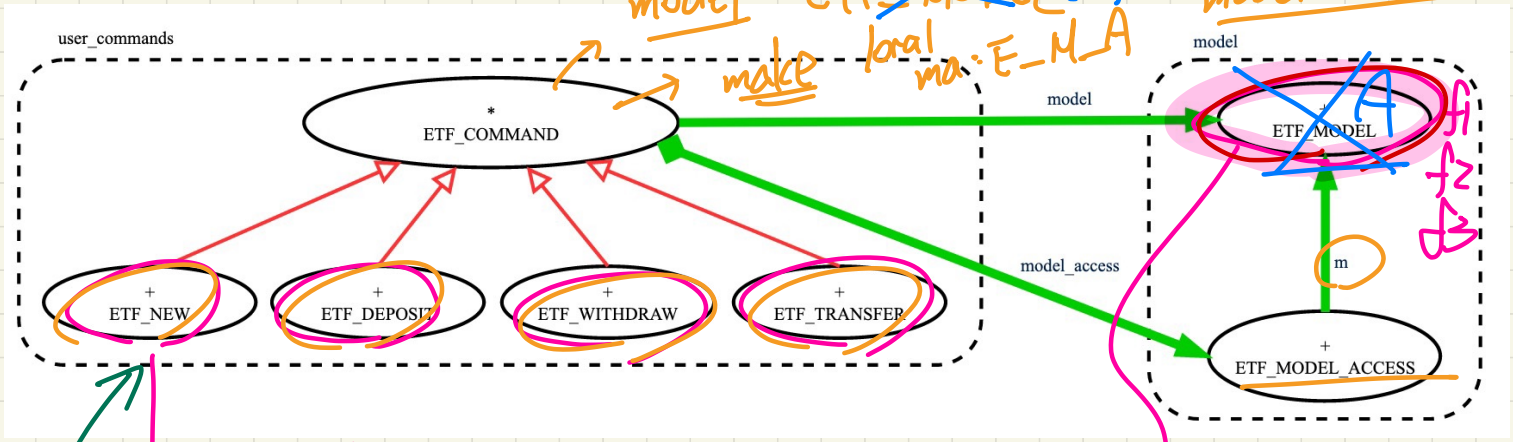
create { NRS } s.make(...)

s.set_dr(0.75)

s : STUDENT

× not allowed by compiler.

s.pr



model : ETF_MODEL_A

make

local ma : E_M_A

model := ma.m.

1. error handling
2. call relevant model operations

model.b.f3

Avoid :

Superman module.

model.

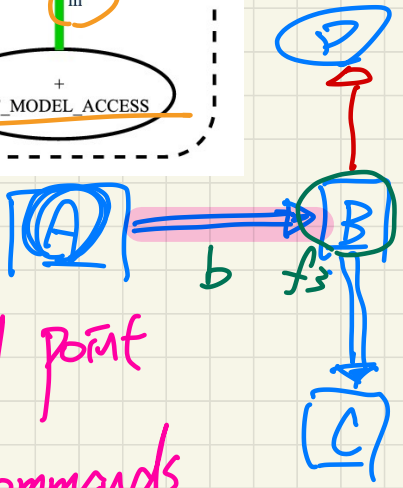
f1
f2
f3

X cohesion

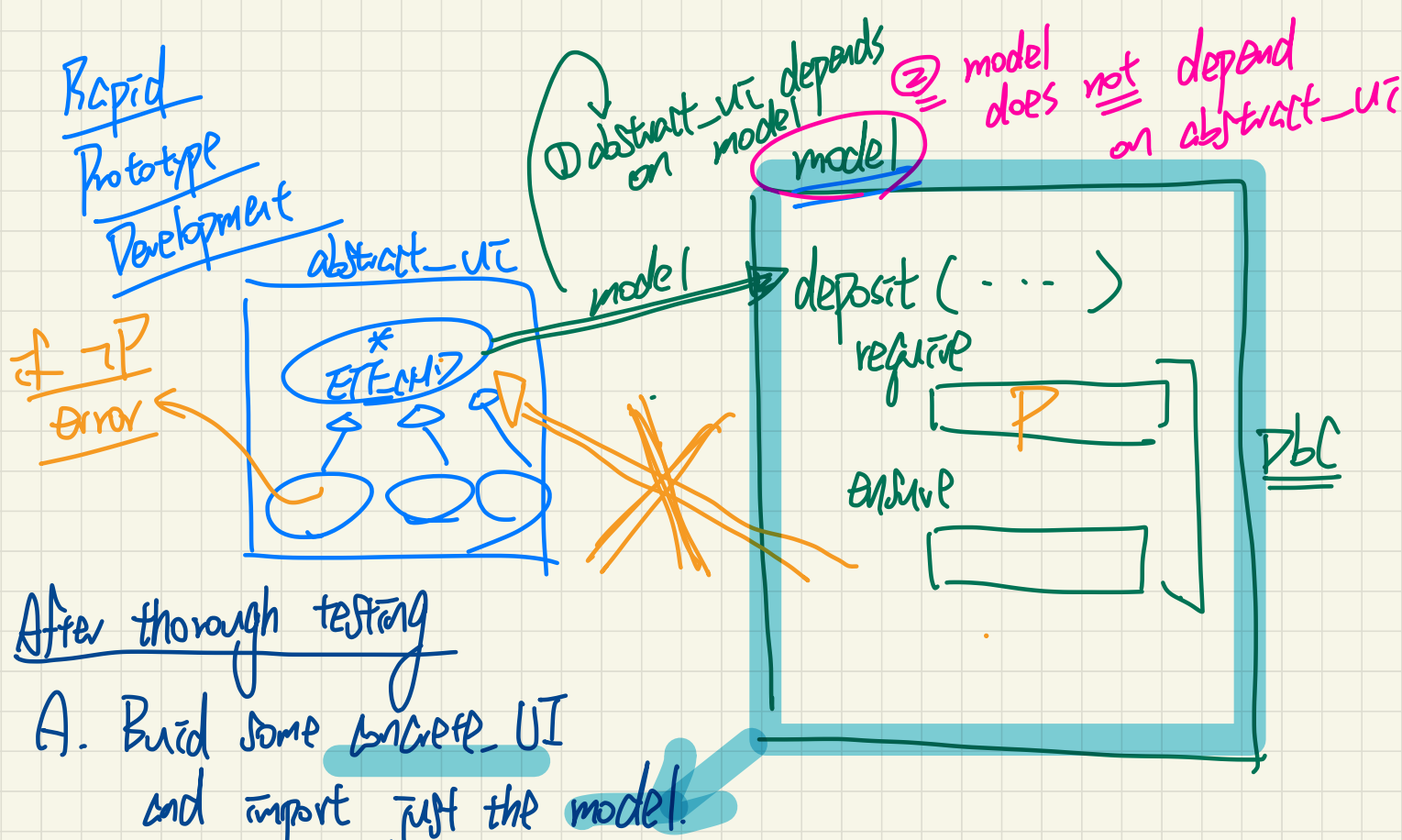
f1
f2
f3
f4
f5
f6

Error messages

single entry point for ETF commands to work with transactions.



Rapid Prototype Development



After thorough testing

A. Build some concrete_UI and import just the model.

B. Include 'model' as a library module.