

In the name of God

SE1 – CA8

Checking bad smells in TinyMe

Some ideas for better designeing

Group 5

Farbod Azimmohseni – 810100187

Amirmahdi Farzane – 810100194

Ali Momtahn - 810100213

Last commit id: f2ee29154bd3b25d7b346a55303dac5aa44aa6ec

Repository: <https://github.com/Farbod82/SE1-Course-Projects>

شناسایی بوهای بد و اعمال بازآرایی

توضیحات (در صورت نیاز)	شناسه کامیت بازآرایی شده	شرح مختصر بوی بد	شناسه کامیت حاوی بوی بد
<p>کلاس orderHandler همزمان valid بود را چک می کند، سپس بر اساس نوع اردر تابع های security را اجرا می کند و در نهایت نتیجه را چک می کند. این ۳ تا کار می تواند به صورت مستقل و در کلاس های دیگری انجام شود</p>	fe587a5e11aa0ed2a0e73ab07ad5e58cf6ce08f5	<p>Large class smell orderHandler کلاس های زیادی انجام می دهد. با move method مشکل حل می شود.</p>	b48056e1e3f6262946b0f3b9c894f5649cc65c50
<p>در کد بازآرایی شده security هیچ وقت به orderhandler و dler eventpublisher دسترسی پیدا نمی کند.</p>	1cacbcfcfaaf25d98c71afae8626e33681bb4d7	<p>اینکه وظیفه handle کردن stop ها به security داده شود این مشکل را ایجاد می کند که دائما security باید به orderhandler و eventpublisher دسترسی پیدا کند که این منطقی نیست و بوی بد Inappropriate Intimacy میگیرد. با استفاده از dependency inversion مشکل حل شد</p>	7352f2b5c8f44fb6a1ef80ffdee2b4ebdfc54c1f
	cf2a051f7da49cafa740443bcd6f5ff20fda39b3	<p>کلاس ChangeMatchState Handler کمی مشکل Dispensable بودن داشت که با refactoring برطرف شد</p>	a715271b132f377546e0d679953b5e85e7a52886
		<p>فعال سازی stopOrder ها بهتر است در خود security انجام شود</p>	1cacbcfcfaaf25d98c71afae8626e33681bb4d7
	<p>4977699c6a104b9dfe7de53e78411a6557db5f4a</p> <p>674369bde9d6e4f7f68ff0af7dc6c21268b98431</p>	<p>در تابع enterOrderRq از security پرسیده می شود که آیا در حالت auction هستی؟ اگر ویژگی های جدید اضافه شود این پراکندگی کنترل</p>	<p>b399f7540080dc45391cd198ccf0a8f66906ecc9,</p> <p>915996e6a24e8b1871b7f7f4f8b29f5723ffec2d</p>

		های auction کار را سخت می کند.	
674369bde9d6e4f7f68ff0af7dc6c2 1268b98431		بعضی از تابع های کلاس orderHandler مشکل Inappropriate Intimacy دارند و متد های آن ها باید به کلاس MatchOutcomePu blisher منتقل شود.	1e26476e25461a0cc79f2824164fec84ef0af844 1cacbcacfcaaf25d98c71afae8626e33681bb4d7
674369bde9d6e4f7f68ff0af7dc6c2 1268b98431		مشکل Change Preventers بین کلاس های orderHandler و MatchOutcomePu blisher کار را برای تغییر سخت می کرد.	1cacbcacfcaaf25d98c71afae8626e33681bb4d7 1e26476e25461a0cc79f2824164fec84ef0af844
674369bde9d6e4f7f68ff0af7dc6c2 1268b98431		مشکل dead code در matchOutcomePu و blisher و orderHandler changeMatchStat eHandler حل شد.	a715271b132f377546e0d679953b5e85e7a52886
قسمتی از کد که تغییرات و credit orderbo را ok انجام میداد از method خارج شده و به تابع دیگری سپرده شد	b48056e1e3f6262946b0f3b9c894f 5649cc65c50	تابع match بسیار طولانی و زیاد است long method	c71867529d13a8707807ed5de2fe6b56e042932a7ad6e5d90176bd32a43b697ce9de8a36448b8197
قسمتی از کد که تغییرات و credit orderbo را ok	d2b3a351cdae913f4874922e1c07 87a7acd18542	طولانی بودن تابع auctionMatch long method	c71867529d13a8707807ed5de2fe6b56e042932a

انجام میداد از method خارج شده و به تابع دیگری سپرده شد			
بازارایی در فایل ها تست	5663e8487e07a7e43dcec5c0eeb5f6f78b66efd4 0ea4e721b674d21be6e08068693ac260a4b26f2e	وجود dead code و duplicate نام گذاری های اشتباه	9cef216242e57156b5460e804bb8df6e84738233
extract method and overload two functions	7ed7d38745de9f9380c08cc61bf8b12ff3951f25	expression passed as input but using function is more readable	8c2c0d26671bc531082b7f6f8264d8d20d474355
extract method	963e81e177cbd493fa8df13b715f2047716dc16a	long method and duplication by similar algorithm	b48056e1e3f6262946b0f3b9c894f5649cc65c50
new constructor for iceberg order	d586ddb05e9c489e7565a725f997c4e521cd996f	Long parameters list I passed enter order request whole object to reduce number of arguments parameters for iceberg order	a7facdb2ca8990ac63156a63a872d48c9ff51683
new constructor for order	4773a502c730793117a5edd8032be6ad79cb9413	Long parameters list I passed enter order request, whole object to reduce number of arguments parameters for order	43c79c3de5d6022fe02204c6d93119b289b6180f
	18826bb003dce6bfed33ed36bc61f20aec566bd3	Long method And assigning multiple tasks to one method	3bbda06ac93d19dab0a9ff0923dfa44a7633b71
Extract method for checking long condition of if statement	037018e29aeceeee7a5f0205c5e6650352d9b3fc	Long method in if statement	d160240f7ea1c0d18acd42f046bf65d240424b64

	bf7df5bd38b705d6b032f8d16b3e2504426a7539	Long method and nested condition statements	b7c969f7c43d66e1479bd5b664af867f12ca8c8b
Extract method for some checking conditions	43c79c3de5d6022fe02204c6d93119b289b6180f	Long method and multiple tasks in one function	a7facdb2ca8990ac63156a63a872d48c9ff51683
Extract more than three functions with unique task from these functions	6afdf94cb0b85778dd6fc5119b68106225dc4ddb	Long methods	43c79c3de5d6022fe02204c6d93119b289b6180f
	2c93ac50089938ac2686948e580a54ca8fee546b	Long method and long condition statement body	6afdf94cb0b85778dd6fc5119b68106225dc4ddb
Extract method from part of if statement	e5be331004de5fef48cd728853219f93d8cf1531	Long method	2c93ac50089938ac2686948e580a54ca8fee546b

موقعیت های بهبود طراحی

۱. هنگام اضافه شدن stopOrder به پروژه، اگر وظیفه handle کردن و publish کردن اردر به security سپرده می شد، security دائم مجبور می شد از متدهای event publisher و orderHandler استفاده کند. با استفاده از dependency inversion، کلاس orderhandler از بالا به security دستورات لازم را می دهد و نتایج را از آن میگیرد و publish می کند. (cacbcafcfaaf25d98c71afae8626e33681bb4d71)

۲. توابع کلاس orderHandler همگن نیستند و بعضی از متود ها validation، بعضی دیگر بررسی نتایج match و بقیه نیز به اختصاص انواع request ها به توابع security می پردازند که این طراحی ایده آلی نیست. برای حل این مشکل، توابع validation و بررسی نتایج match به دو کلاس utility منتقل شدند. (fe587a5e11aa0ed2a0e73ab07ad5e58cf6ce08f5)

۳. در فاز قبل و هنگامی که auction match به پروژه اضافه شد در خیلی از کلاس ها از جمله orderHandler شرط های اضافه ای در هنگامی که security در حالت auction است به وجود آمد. در این refactoring، وابستگی orderHandler به auctionMatch تقریبا از بین رفت و تقریبا تمام check های لازم در جایی که واقعا به آن مربوط است یعنی security بررسی می شوند. (c6a104b9dfe7de53e78411a6557db5f4a۹۷۷۶۹۹)

۴. انتقال بررسی exception برای بروزرسانی که به کلاس order validator انتقال داده شد.

۵. ساخت یک کلاس control برای چک کردن موارد زیاد credit checking در کلاس matcher وجود دارد و لزوما وظیفه matcher نیست (f2ee29154bd3b25d7b346a55303dac5aa44aa6ec)

شناسه کامیت پیاده سازی موقعیت های بهبود طراحی : f2ee29154bd3b25d7b346a55303dac5aa44aa6ec