Métricas

APH (Árbol de Profundidad de Herencia) = 2.

Todas nuestras clases Extienden de Model, generando únicamente descendientes directos. Esta métrica representa a través de un valor, la profundidad del árbol de herencias, al tener un bajo APH estamos reutilizando menos código, pero tenemos un diseño más limpio, las operaciones y atributos heredadas por nuestras clases son acotadas, lo cual no complejiza los Tests de las mismas, ni las torna impredecibles.

NDD (Numero de Descendientes Directos) = 4.

Al extender nuestras clases de MODEL, estamos generando descendientes directos y reutilizamos código, al no ser un número tan alto, no se pierde la abstracción y los descendientes siguen siendo buenos representantes de la misma.

ACO (Acoplamiento) USER =2

ACO GAME = 3

ACO CATEGORY = 1

ACO QUESTION = 1

ACOAPP = 4

Estas métricas fueron obtenidas del diagrama de clases del Preguntado\$ diseñado en Análisis y Diseño de Sistemas. El acoplamiento mide la cantidad de colaboradores de una clase, un número alto de acoplamiento hace más complicado modificar el comportamiento de una clase ya que puede tener grandes efectos colaterales. Nuestra clase App.java es la que se comunica con todas las demás, por eso el ACO fue mayor.

TC (Tamaño de Clase, por METODOS PROPIOS)

TC USER = 8

(answerQuestion(),getGameInProgress(),win_rate(),c_questions(),userName(),createGameforUser(),
user(), user())

Un TC alto genera una clase más específica de la aplicación, disminuyendo su reutilización

NOR QUESTION (Número de Operaciones

Redefinidas) = 0

NOR GAME = 0

NOR CATEGORY = 0

NOR USER = 0

NOR APP = 0

Un número alto de operaciones redefinidas significa que la herencia es débil.

NOA = TC (En este caso porque con contamos los métodos heredados en el TC)

NP media QUESTION = 10/2 = 5

NP media CATEGORY = 3/4

NP media GAME = 4/5

NP media USER = 11/8

NP media APP = 22/11 = 2