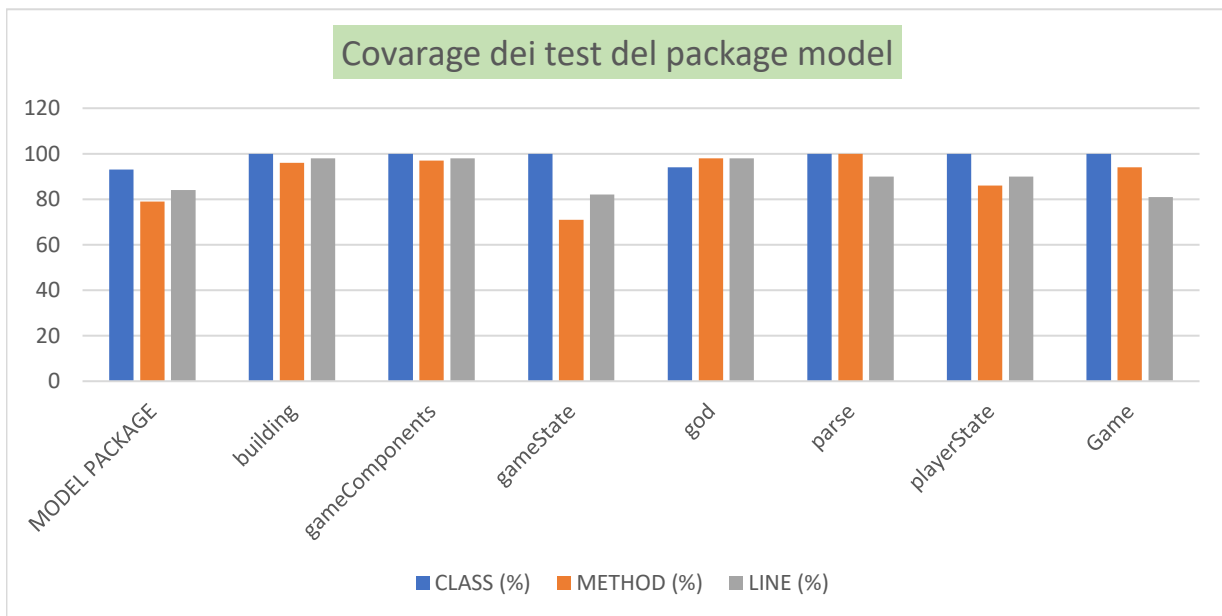


Covarage dei test

	CLASS (%)	METHOD (%)	LINE (%)
MODEL PACKAGE	93	79	84
building	100	96	98
gameComponents	100	97	98
gameState	100	71	82
god	94	98	98
parse	100	100	90
playerState	100	86	90
Game	100	94	81



	CLASS	METHOD	LINE
MODEL PACKAGE	45 of 48	397 of 497	1195 of 1414
building	11 of 12	32 of 35	81 of 89
gameComponents	4 of 4	79 of 81	221 of 224
gameState	4 of 4	35 of 49	71 of 87
god	18 of 19	187 of 192	490 of 509
parse	1 of 1	6 of 6	79 of 87
playerState	6 of 6	36 of 42	72 of 79
Game	1 of 1	48 of 50	190 of 227

I test sono stati eseguiti principalmente sul Model, ovvero su una delle componenti del pattern MVC seguendo una logica di copertura degli statement. Questo ci ha portato ad avere una buona percentuale di copertura delle istruzioni di ciascun metodo, anche se in alcuni casi abbiamo preferito non testare metodi come "getter" e "setter" dei parametri. Abbiamo deciso di non testare la classe ProxyGameModel in quanto quest'ultima è solo una classe che astrae alcuni dei metodi della classe Game per separare la parte di logica di gioco pura dal resto. infatti il ProxyGameModel sarà la classe richiamata dal controller e dalla virtualview del pattern MVC. Grazie ad una percentuale alta di coverage siamo riuscite ad avere sempre sotto controllo eventuali problei relativi all'implementazione e a testare singolarmente le classi tanto da avere una buona struttura modulare con componenti autonomi tra loro.