

# Blind Trust

Piattaforma anonimizzante per gestione di concorsi e selezioni pubbliche



# Introduzione

Gare d'appalto e concorsi pubblici truccati, bandi creati su misura per far sì che il vincitore sia stabilito *ex ante*; chiunque segua l'attualità e la cronaca italiana saprà che situazioni di questo tipo sono all'ordine del giorno, e non si vedono accenni di diminuzione del fenomeno.

Nel corso degli anni non sono certo mancati i tentativi di riformare la pubblica amministrazione (nelle sue varie componenti) per cercare di arginare questi fenomeni. Nel 2009 la riforma Gelmini prometteva la fine delle baronie negli atenei e stabiliva criteri che -sempre secondo i proponenti- avrebbero garantito trasparenza e meritocrazia. Più di recente l'istituzione dell'[ANAC](#) nel 2102, il [nuovo codice degli appalti](#) nel 2016 e diverse altre misure hanno quantomeno tentato di porre rimedio a quella che viene da molti considerata la principale piaga nazionale.

Eppure, le pagine di cronaca giudiziaria continuano a mostrare uno scenario apparentemente impermeabile a qualunque tentativo di riforma.

Il progetto Blind Trust si propone quindi di risolvere il problema intervenendo alla radice: partendo dal presupposto che il comune denominatore delle situazioni sovraccitate (gare d'appalto, concorsi per ricercatori universitari etc.) è **la presenza di una giuria giudicante e una platea di giudicati**, e che i favoritismi avvengono quasi sempre perché chi giudica conosce ed è in combutta con un giudicato, si propone di creare una piattaforma che **anonimizzi il maggior numero possibile di informazioni**, in modo che la valutazione si focalizzi esclusivamente sui contenuti e sia meno influenzata possibile dall'identità del candidato.

Questo tipo di prassi è peraltro già ampiamente utilizzata in altri contesti, come i concorsi letterari: tipicamente ai partecipanti viene chiesto di inviare gli elaborati in più copie (solitamente una per ogni membro della giuria), di cui soltanto una recante le generalità del concorrente. Ai giurati viene sottoposto il testo non firmato, così da evitare condizionamenti di sorta.

## Sommario

L'obiettivo è implementare il prototipo di una generica piattaforma di valutazione che garantisca "by design" e senza possibilità di frode, mediante metodo del doppio cieco, la raccolta dei test dei candidati e la selezione dei vincitori della gara. Questa piattaforma potrà servire, ad esempio, per valutare prove di ammissione, certificazioni, abilitazioni e concorsi.

## Proof of Concept: concorso per ricercatori università

Una situazione in cui un principio analogo viene già parzialmente usato è rappresentata dai concorsi letterari: tipicamente ai partecipanti viene chiesto di inviare gli elaborati in più copie (solitamente una per ogni membro della giuria), di cui soltanto una recante le generalità del concorrente. Ai giurati viene sottoposto il testo non firmato, così da evitare condizionamenti di sorta.

Rispetto ai concorsi letterari amatoriali, tuttavia, un concorso per ricercatori universitari assume connotati assai più complessi. Oltre agli elaborati, occorrerebbe anonimizzare anche altre informazioni, e segnatamente:

- gli aspiranti ricercatori non dovrebbero conoscere l'identità dei giudicanti
- i membri della giuria non dovrebbero conoscersi reciprocamente
- i membri della giuria non dovrebbero conoscere il “contesto” per il quale sono stati chiamati ad esprimere un giudizio

In concreto:

1. L'Università indice un bando per  $X$  posti di ricercatore. Il/i posto/i verranno assegnati sulla base della valutazione di un elaborato scritto che avrà per oggetto un determinato tema. L'accesso al concorso sarà legato all'aver conseguito un tot di CFU in determinate aree disciplinari (vd. Sotto per approfondimento).
2. Il candidato in possesso dei requisiti accede al sito del proprio Ateneo e in seguito a Blind Trust, e redige l'elaborato.
3. La piattaforma Blind Trust seleziona, su base nazionale e semi-randomica, cinque membri giudicanti, sulla base di pubblicazioni, competenze e altri criteri (vd. sotto).
4. Ai docenti prescelti arriverà via mail una richiesta di giudizio in cui verrà mostrato soltanto il tema e gli elaborati. Il docente sarà chiamato ad assegnare una valutazione a ciascuno di essi entro un determinato lasso di tempo.
5. Al termine della procedura viene pubblicata la graduatoria dei partecipanti e rivelato quale Ateneo ha bandito il bando. I primi in graduatoria decideranno se accettare o meno, e i posti verranno assegnati seguendo la graduatoria stessa.

## **Elenco dei requisiti BlindTrust**

Tale piattaforma dovrà:

1. predisporre autonomamente i test da sottoporre ai candidati in base alla difficoltà richiesta dalla gara.
2. raccogliere gli elaborati e i test forniti dai candidati anonimizzati.
3. esaminare e controllare autonomamente gli esiti dei test o assegnare la valutazione degli stessi agli esaminatori in base ad un principio di doppio cieco.
4. assegnare un punteggio ai test calcolato in base ad un algoritmo pubblico e rendere verificabili gli esiti dei test.
5. assicurare la non modificabilità dei contenuti dei test / elaborati una volta “consegnati”

In questa fase prototipale implementeremo un format di test base, del tipo “tema”, da distribuire per valutazione agli esaminatori.

Saranno pertanto esclusi, dal prototipo, test di valutazione a correzione automatica (ad esempio test con risposta da barrare tra un pool di risposte date).

## Architettura della piattaforma

La piattaforma è composta da un modulo core dedicato all'espletamento delle principali funzionalità, ovvero la verifica e lo smistamento degli elaborati agli esaminatori, la creazione dei test e la gestione del flusso delle informazioni tra gli attori in gioco che sono:

1. Editor: scrivono e inseriscono nel sistema temi/test/quiz ai quali devono assegnare un punteggio di difficoltà e un'area tematica di pertinenza. Tutte queste informazioni sono inserite in un database per essere disponibili durante la creazione della gara.
2. Esaminatori: sono coloro che verranno chiamati ad esaminare gli elaborati che non possono essere valutati in automatico dal sistema.
3. Candidati: è l'insieme degli utenti iscritti per superare il concorso/test.

## Tassonomie

L'elenco delle tassonomie delle aree associate ai quiz/temi.

Per la versione prototipo ci limiteremo ad una tassonomia ridotta ai termini essenziali per un funzionamento base.

## Gestione flusso

Un esempio di flusso in un caso reale

- L'ente appaltante apre e configura il concorso, assegna il titolo, il periodo di validità, l'elenco dei requisiti, un livello di difficoltà o predisporre il test in base a livelli di difficoltà variabili.
- il sistema genera il test assemblando le eventuali domande a risposta multipla o le eventuali tracce dei problemi e temi estratti dal database in base ai vincoli e ai requisiti, tenendo conto del livello di difficoltà puntuale e globale.
- il sistema pubblica il test al quale partecipano le persone iscritte (che ne hanno titolo)
- il sistema raccoglie i test, li anonimizza e li corregge o li distribuisce per la valutazione agli esaminatori
- gli esaminatori valutano gli elaborati degli utenti e attribuiscono i punteggi. Ogni elaborato viene esaminato da almeno tre esaminatori. Il voto più alto e quello più basso vengono scartati. Gli altri contribuiscono al calcolo della media del punteggio finale.
- Il sistema ordina i test in base ai punteggi conseguiti, verifica che l'hash non sia modificato e convalida i risultati assegnandoli ai legittimi autori.
- I dati vengono resi disponibili per verifiche a posteriori anche da terze parti.
- Il sistema controlla e segnala eventuali anomalie, come significativi scostamenti dalla media nelle valutazioni degli esaminatori.

## Tipologie utenti

Predisponiamo nell'attivazione del modulo le seguenti categorie di utenza.

### **Editor**

Gli editor inseriscono i quiz, i test, le tracce dei temi, i problemi da sottoporre ai candidati. Disporranno di una interfaccia di inserimento, con possibilità di inserire testi liberi, disegni, e predisporre test a risposta multipla (ma non nel caso del prototipo per il quale scegliamo una versione semplificata). Dovranno anche assegnare un punteggio in base alla difficoltà del quiz / test.

### **Candidato**

Si iscrivono al concorso entro i tempi stabiliti dallo stesso. Partecipano ai test che, al submit viene criptato ed hashato per verificarne l'integrità al momento della correzione (garantendo così che il test non potrà essere manomesso durante le fasi successive).

### **Esaminatore**

Vengono selezionati casualmente dal sistema per valutare ciò che il sistema non può valutare in maniera automatica. Ogni elaborato dovrà essere valutato da un numero di esaminatori  $\geq 3$  per permettere il calcolo medio del punteggio una volta esclusi i punteggi massimi e minimi.

## Task

Il sistema dovrà assicurare alcune funzionalità

### **Controlli sulle verifiche**

Il sistema rileverà lo scarto dalla media di valutazione dei test e, oltre una determinata soglia, solleverà un alert sugli esaminatori che si distanziano eccessivamente dalla media delle valutazioni.

### **Generare i test**

Il generatore è il software che costruisce il concorso assemblando le infrazioni che provengono dal database in base alla difficoltà richiesta e alle aree di contenuto.

Il testo del concorso verrà pubblicato solamente all'inizio della gara.

## Framework di sviluppo

Drupal 8 (o alternative da valutare)

## Eventuale

Generazione dei moduli Drupal

Si richiede la generazione di un modulo Drupal che

- crei le tipologie di utenza come descritte sopra
- installi il core che contiene
- operazioni di selezione casuale degli esaminatori

- calcoli il punteggio dei test
- calcolo della lista dei vincitori
- assemblaggio l test in base ad un livello di difficoltà

Al modulo core andrà agganciato un modulo in grado di implementare vari tipi di test, (es. quiz, tema ecc...)