



# Dashboard Streamlit per il Calcolo del Multidimensional Risk Score

## Sommario

1. Descrizione del Progetto .....	2
2. Struttura del Codice .....	2
2.1 Descrizione dei dati .....	2
2.2 Caricamento dati .....	2
2.3 Descrizione modulo tools.py .....	3
2.4 Autenticazione .....	3
2.5 Interfaccia.....	3
2.6 Visualizzazioni.....	3
2.7 Download Risultati.....	4
3. Guida al deploy su Streamlit Cloud.....	5
3.1 Requisiti preliminari .....	5
3.2 Deploy su Streamlit Cloud .....	5
3.3 Configurazione dei secrets su Streamlit Cloud .....	5
3.4 Testing e aggiornamenti .....	5
4. Suggerimenti per l'Integrazione di Nuove Funzionalità .....	7
4.1 Integrazione di nuove fonti dati.....	7
5. Possibili Estensioni e Miglioramenti .....	8
5.1 Espansione dei contenuti .....	8
5.2 Potenziamento delle funzionalità.....	8
5.3 Esperienza utente e accessibilità .....	8
5.4 Scalabilità della piattaforma .....	8
5.5 Suggerimenti basati sul punteggio di rischio .....	9

# 1. Descrizione del Progetto

Questa dashboard interattiva, sviluppata in **Streamlit**, permette di calcolare il **Multidimensional Risk Score** di un Paese target sulla base di:

- **Settore aziendale**
- **Strategia di ingresso**
- **Profilo di rischio del Paese**

La piattaforma consente di:

- Selezionare dinamicamente Paese, Settore e Strategia.
- Visualizzare grafici (bar chart, radar chart).
- Scaricare un **report Excel dettagliato** del calcolo.
- Proteggere l'accesso tramite **login con password**.

## 2. Struttura del Codice

### 2.1 Descrizione dei dati

I dati in riferimento al file "Scoring\_Rischio\_Internazionale\_1.xlsx", in questa prima versione sono suddivisi in tre sheet, sottoforma di tabella con struttura simile. Questo formato permette una lettura immediata e semplificata delle informazioni necessarie.

### 2.2 Caricamento dati

File utilizzato: Scoring\_Rischio\_Internazionale\_1.xlsx

Funzioni di caricamento (dal modulo tools.py):

- caricamento\_dati\_fattori
- caricamento\_dati\_pesi
- caricamento\_dati\_rischio

Questi dati contengono:

- Pesi associati ai rischi per ogni settore.
- Fattori di moltiplicazione per ciascuna strategia.
- Valori di rischio (Politico, Economico, Legale, ecc.) per ciascun Paese.

Per aggiunte/ modifiche di dati alla dashboard:

- Per aggiungere paesi, inserire righe nel file " Scoring\_Rischio\_Internazionale\_1.xlsx " , nel foglio "risk".
- Per aggiungere nuove dimensioni di rischio, inserire colonne file " Scoring\_Rischio\_Internazionale\_1.xlsx " , nel foglio "risk".

- Per aggiungere nuovi settori, inserire righe nel file nel file “ Scoring\_Rischio\_Internazionale\_1.xlsx “ , nel foglio “pesi\_settore”.
- Per aggiungere nuove strategie, inserire righe nel file nel file “ Scoring\_Rischio\_Internazionale\_1.xlsx “ , nel foglio “fattori”.
- Per modificare i valori attuali, agire sul file “ Scoring\_Rischio\_Internazionale\_1.xlsx “

Note strutturali:

- Qualsiasi modifica effettuata sul file “ Scoring\_Rischio\_Internazionale\_1.xlsx “, impatterà sulle visualizzazioni della dashboard online.
- Aggiunte al file “ Scoring\_Rischio\_Internazionale\_1.xlsx “ , richiederanno un adattamento degli script “app.py” e “tools.py”.

## 2.3 Descrizione modulo tools.py

Con il seguente modulo si creano funzioni per la lettura dei dati dal file excel principale. In particolare, ogni informazione è letta dalle varie sheet presenti nel file di riferimento. Per ogni lettura che viene fatta, i dati vengono salvati in un dizionario chiave-valore, che saranno poi utilizzati nel app.py per il calcolo dello score e la creazione dei grafici

## 2.4 Autenticazione

- La dashboard è protetta da password (st.secrets["DASHBOARD\_PASSWORD"]), che è gestita tramite le funzionalità di streamlit cloud, in particolare nella sezione dedicata alle impostazioni
- L'utente deve effettuare il login nella **sidebar**.
- Il logo viene mostrato prima e dopo il login (logo3clab.png).

## 2.5 Interfaccia

- Selezione del **Paese target**, del **settore** e della **strategia di ingresso**.
- Calcolo di uno score totale personalizzato, basato su:
  - Peso del settore per ciascun tipo di rischio.
  - Fattore strategico.
  - Valore del rischio nel Paese target.

Funzione principale: def calcola\_score\_nuovo(rischi, pesi, fattori)

## 2.6 Visualizzazioni

- **Bar Chart**: mostra il valore finale per ciascun tipo di rischio.
- **Score Totale**: metrica personalizzato.
- **Classifica Paesi**: in base allo score calcolato.
- **Radar Chart**: confronto visivo tra i Paesi su 5 dimensioni (Politico, Economico, Culturale, Legale, Innovazione)

## 2.7 Download Risultati

- L'utente può scaricare un file .xlsx con tutti i dettagli del calcolo.
- La tabella include anche una **riga finale con il totale**.

## 3. Guida al deploy su Streamlit Cloud

Questa sezione descrive come eseguire il deploy della dashboard su Streamlit Cloud, rendendola accessibile da browser tramite un semplice link.

### 3.1 Requisiti preliminari

- Un account su **GitHub**
- Un account su **Streamlit Cloud**
- Il progetto salvato in un repository GitHub, con i seguenti file principali:
  - app.py (o qualunque sia lo script principale)
  - requirements.txt (contenente tutte le librerie necessarie)
  - La cartella tools/ (con i file Python richiamati nel codice)
  - Il file Excel o in generale il file contenente i dati usati (Scoring\_Rischio\_Internazionale\_1.xlsx)
  - Immagini o altri asset usati (logo3clab.png, ecc.)

### 3.2 Deploy su Streamlit Cloud

- Accedere <https://streamlit.io/cloud> con account GitHub
- “**New app**”
- Selezionare il repository GitHub e il branch (es. main)
- Inserire il nome del file Python da lanciare (es. dashboard.py)
- Infine, su “**Deploy**”

### 3.3 Configurazione dei secrets su Streamlit Cloud

Per aggiungere e cambiare le variabili segrete (come la password), andare alla dashboard Streamlit Cloud:

- Cliccare su app > **Settings** > **Secrets**
- Inserire una variabile locale segreta come in questo caso  
“DASHBOARD\_PASSWORD = "la\_tua\_password\_sicura”

### 3.4 Testing e aggiornamenti

Una volta avviata l'app:

- Accesso tramite il link fornito da Streamlit Cloud
- Per **ogni push su GitHub**, l'app si aggiornerà automaticamente

- Per problemi di caricamento o errori, consultare i **Logs** dalla dashboard Streamlit Cloud

## 4. Suggerimenti per l'Integrazione di Nuove Funzionalità

Per estendere o aggiornare la dashboard in futuro, si consiglia di:

- **Seguire la struttura dati esistente:** qualunque nuovo file Excel o fonte dati da aggiungere dovrebbe rispettare la stessa struttura utilizzata nella demo attuale specificando, nomi dei fogli, formato delle righe e colonne, naming delle dimensioni delle tabelle, ecc. Questo approccio garantisce compatibilità e integrazione immediata nel flusso di calcolo.
- **Modularizzare il codice:** eventuali nuove funzionalità (es. calcoli personalizzati, nuove visualizzazioni, filtri avanzati) dovrebbero essere implementate in **script di supporto separati**, all'interno della cartella tools/ o simili, per evitare di appesantire il file principale della dashboard e mantenerlo leggibile e manutenibile.

### 4.1 Integrazione di nuove fonti dati

Per estendere la copertura informativa e migliorare l'analisi del rischio, è possibile integrare nuove fonti di dati nella dashboard. Di seguito sono indicati i passaggi consigliati per farlo in modo strutturato, scalabile e coerente con l'architettura attuale.

#### 1. Creare script di supporto dedicati

- Per mantenere il codice della dashboard snello, la logica di parsing, pulizia e validazione dei nuovi dati dovrebbe essere scritta in **moduli ausiliari** (es. data\_loader.py, data\_cleaning.py).
- Ogni nuova fonte può essere gestita come una funzione a sé stante, facilmente testabile e riutilizzabile.

#### 2. Validazione dei dati

- Implementare **controlli di coerenza** sui nuovi dataset (es. range di valori, colonne obbligatorie, formato standardizzato).
- È consigliato predisporre una funzione di **verifica automatica del formato** prima di caricare i dati nella dashboard.

#### 3. Supporto per fonti online

- Per aumentare l'automazione, è possibile aggiungere **connessioni a fonti esterne** tramite API pubbliche o scraping controllato.
- In questi casi, usare script separati per il fetch e l'aggiornamento periodico dei dati.

## 5. Possibili Estensioni e Miglioramenti

### 5.1 Espansione dei contenuti

- **Fonti dati dinamiche:** Integrare fonti dati aggiornabili automaticamente (es. API della Banca Mondiale, IMF, World Economic Forum) per mantenere sempre attuali gli indicatori.
- **Confronto storico:** Permettere il confronto dei punteggi di rischio nel tempo, se sono disponibili dati storici.

### 5.2 Potenziamento delle funzionalità

- **Personalizzazione avanzata dei pesi:** Dare all'utente la possibilità di definire manualmente i pesi dei rischi o salvare preset personalizzati.
- **Simulatore di scenari:** Integrare un tool interattivo che permetta di simulare variazioni nei fattori di rischio o nelle strategie per valutare impatti potenziali sul punteggio.
- **Benchmarking settoriale:** Consentire il confronto tra paesi per uno specifico settore, visualizzando anche la media globale o regionale.

### 5.3 Esperienza utente e accessibilità

- **Interfaccia multilingua:** Supportare l'uso in diverse lingue (es. inglese, spagnolo), utile per un utilizzo internazionale.
- **Esportazione automatica reportistica:** Generare report PDF automatici personalizzati per paese/settore/strategia con tabelle e grafici già pronti.

### 5.4 Scalabilità della piattaforma

#### 1. Separazione dei moduli

- Mantenere lo script principale (app.py) snello, spostando funzioni, logiche di calcolo e gestione dei dati in **moduli separati** (es. tools.py, utils.py, services/).
- Questo migliora la **manutenibilità** e consente aggiornamenti mirati senza impattare l'intera interfaccia.

#### 2. Gestione dei dati dinamica

- Collegare la dashboard a **database remoti** (es. PostgreSQL, MongoDB, Google Sheets, etc.) al posto di file Excel statici, per permettere aggiornamenti **in tempo reale** e gestione multiutente.
- In alternativa, integrare un sistema di **upload dati** lato utente (con validazione e controllo del formato).

#### 3. Autenticazione avanzata



- Implementare un sistema di login con **ruoli e permessi** (es. amministratori, utenti base) tramite servizi OAuth (Google, GitHub) o librerie come streamlit-authenticator.
- Questo rende possibile personalizzare la visualizzazione e i dati accessibili per ciascun utente.

#### 4. Hosting professionale

- Per esigenze avanzate, è consigliabile migrare da Streamlit Cloud a soluzioni più robuste:
  - **Docker + cloud provider (AWS, GCP, Azure)** per il pieno controllo dell'infrastruttura
  - **Streamlit sharing + CDN** per bilanciare il carico in caso di molti utenti contemporanei

### 5.5 Suggerimenti basati sul punteggio di rischio

Un possibile sviluppo futuro della dashboard consiste nell'integrazione di un **sistema di raccomandazione intelligente**. In base allo **score totale di rischio** calcolato per ciascuna combinazione Paese–settore–strategia, la piattaforma potrebbe fornire **consigli operativi** personalizzati.

#### Esempi di suggerimenti:

- Se il punteggio di rischio è **elevato**, suggerire strategie più conservative (es. joint venture, presenza indiretta).
- Se il punteggio è **medio**, consigliare ulteriori analisi qualitative o fonti aggiuntive prima di decidere.
- Se il rischio è **basso**, proporre modalità di ingresso più dirette o investimenti più consistenti.

#### Questo modulo potrebbe essere realizzato tramite:

- Una **logica a regole**, inizialmente semplice e interpretabile.
- In una fase avanzata, l'uso di modelli **ML-based** (es. classificatori supervisionati) allenati su casi storici di internazionalizzazione.