

# Deliverable 3

# CBAM

---

GABRIELE QUATRANA 0306403

# Sommario

- Introduzione
- Decisione
- Attributi di Qualità
  - Java
  - C++
  - C
  - Python
- Rischio
- Beneficio
- Costo
- Desiderabilità
- Classifica
- Conclusioni

# Introduzione

---

- Si vuole sviluppare un software, desktop e web, per la gestione della vita universitaria di studenti e professori chiamato *.myUniversity*.
- Il programma deve mettere a disposizione degli utenti diverse funzionalità:
  - Uno studente può registrarsi ad un corso inviando una richiesta al professore.
  - Uno studente può tenere traccia dei voti ricevuti.
  - Uno studente può visualizzare le prossime lezioni relative ai corsi a cui si è iscritto.
  - Uno studente può formulare delle domande da porre ad altri studenti o al professore del corso.
  - Un professore può accettare o rifiutare una richiesta di registrazione ad uno dei suoi corsi.
  - Un professore può organizzare le lezioni future e gli esami per i corsi che gestisce.

# Introduzione

---

- Per definire l'architettura software del sistema si vuole utilizzare la tecnica **CBAM** (Cost Benefit Analysis Method).
- Il **CBAM** è una tecnica decisionale che si basa sui **costi** e i **benefici** delle varie alternative:
  - Una tecnica decisionale è una tecnica che permette di selezionare la migliore delle alternative per una decisione.

# Decisione

---

- La decisione considerata (il problema da risolvere) riguarda la scelta del linguaggio di programmazione da utilizzare per implementare l'applicazione.
- Sono state individuate quattro alternative:
  - **Java**
  - **C++**
  - **C**
  - **Python**

# Attributi di Qualità

---

- Gli attributi di qualità considerati sono i seguenti:
  - **Performance:** indica quanto il programma è veloce nell'eseguire le operazioni in base al linguaggio con cui è stato sviluppato.
  - **Security:** indica quanto il linguaggio sia privo di vulnerabilità.
  - **Modifiability:** indica la semplicità con la quale è possibile apportare modifiche al codice.
  - **Availability:** indica il periodo di tempo nel quale il programma è disponibile, rispetto al tempo totale in cui è attivo, in base al linguaggio con cui è stato implementato.
  - **Interoperability:** indica la capacità del linguaggio di interagire con sistemi esterni.
  - **Integrability:** indica la capacità del linguaggio di interagire con altri linguaggi di programmazione.

# Attributi di Qualità

---

- Associamo ad ogni attributo un peso in base alla loro importanza e in modo tale che la somma dei vari pesi sia pari a 100:

○ Performance	25
○ Security	20
○ Modifiability	20
○ Availability	6
○ Interoperability	17
○ Integrability	12

# Java

---

- Performance 0.0
- Security 0.9
- Modifiability 0.7
- Availability 0.8
- Interoperability 0.6
- Integrability 0.5

- Pro:
  - Java è molto semplice da utilizzare, anche grazie all'allocazione automatica di memoria e al garbage collection.
  - È indipendente dalla piattaforma di esecuzione, grazie all'utilizzo del byte-code e della JVM.
  - È molto sicuro in quanto non prevede l'utilizzo di pointer espliciti.
  - Molto utilizzato per lo sviluppo di codice lato server.
- Contro:
  - Java consuma molta memoria ed è significativamente più lento rispetto ad altri linguaggi.
  - In Java la memoria viene gestita tramite il garbage collector e l'esecuzione di questo thread impatta le performance dell'applicazione.



# C++

---

- Performance 0.9
- Security -0.2
- Modifiability 0.4
- Availability 0.7
- Interoperability 0.5
- Integrability 0.6

- Pro:
  - Offre un'alta portabilità.
  - È compatibile con il linguaggio C.
  - È altamente scalabile.
  - Ha una community molto sviluppata che fornisce un buon supporto.
- Contro:
  - Problemi di sicurezza dovuti alla presenza dei puntatori.
  - Linguaggio molto complesso da padroneggiare.
  - Assenza del garbage collector.

# C

---

- Performance 0.9
- Security -0.6
- Modifiability 0.4
- Availability 0.6
- Interoperability 0.5
- Integrability 0.5

- Pro:
  - Offre un'alta portabilità.
  - Linguaggio robusto ed efficiente, grazie anche ai molti tipi di dato e operatori che fornisce di base.
  - Gestione dinamica della memoria.
- Contro:
  - Problemi di sicurezza dovuti all'utilizzo dei puntatori e al basso livello di astrazione.
  - Non è presente un sistema di gestione delle eccezioni.

# Python

---

- Performance -0.3
- Security 0.8
- Modifiability 0.9
- Availability 0.5
- Interoperability 0.8
- Integrability 0.7

- Pro:
  - Estremamente versatile e facile da usare.
  - Ha una community molto sviluppata che rappresenta un buon supporto.
  - Semplice da integrare con altri sistemi.
  - Ampiamente utilizzato per lo sviluppo di applicazioni web.
- Contro:
  - Significativamente più lento rispetto ad altri linguaggi di programmazione.
  - Il consumo di memoria da parte di applicazioni Python è molto elevato.

# Rischio

---

- $Risk(Java) = 0.0$ 
  - Java ha un rischio nullo, in quanto il team di sviluppo ha molta esperienza con tale linguaggio, anche nel trattare applicazioni web. Inoltre, l'ampia documentazione presente permette di gestire lo sviluppo delle funzionalità più complesse.
- $Risk(C++) = 0.5$ 
  - C++ ha un rischio moderato, in quanto il team di sviluppo non ha nessuna esperienza con tale linguaggio, ma l'ampia documentazione presente permette di padroneggiarlo in tempi abbastanza brevi.
- $Risk(C) = 0.7$ 
  - C ha un rischio alto, in quanto è altamente sconsigliato per lo sviluppo di applicazioni web a causa dei problemi di sicurezza che presenta.
- $Risk(Python) = 0.3$ 
  - Python ha un rischio basso, in quanto il team di sviluppo ha avuto esperienze pregresse con tale linguaggio, ma non riguardo lo sviluppo di applicazioni web.

# Beneficio

---

$$Benefit(AS_i) = \sum_{j=1}^6 ((AS_{ij} * QAscore_j)) * |Risk_i - 1|$$

$$Benefit(Java) = (25 * 0.0 + 20 * 0.9 + 20 * 0.7 + 6 * 0.8 + 17 * 0.6 + 12 * 0.5) * (1 - 0.0)$$

$$Benefit(Java) = 53.0$$

$$Benefit(C++) = (25 * 0.9 + 20 * (-0.2) + 20 * 0.4 + 6 * 0.7 + 17 * 0.5 + 12 * 0.6) * (1 - 0.5)$$

$$Benefit(C++) = 23.2$$

$$Benefit(C) = (25 * 0.9 + 20 * (-0.6) + 20 * 0.4 + 6 * 0.6 + 17 * 0.5 + 12 * 0.5) * (1 - 0.7)$$

$$Benefit(C) = 10.98$$

$$Benefit(Python) = (25 * (-0.3) + 20 * 0.8 + 20 * 0.9 + 6 * 0.5 + 17 * 0.8 + 12 * 0.7) * (1 - 0.3)$$

$$Benefit(Python) = 36.05$$

# Costo

---

- Tutti i linguaggi di programmazione considerati sono disponibili gratuitamente, quindi il costo delle varie alternative è pari ad 1:
  - $Cost(Java) = 1$
  - $Cost(C++) = 1$
  - $Cost(C) = 1$
  - $Cost(Python) = 1$

# Desiderabilità

---

$$Desiderability(AS_i) = \frac{Benefit(AS_i)}{Cost(AS_i)}$$

$$Desiderability(Java) = \frac{Benefit(Java)}{Cost(Java)} = 53.0$$

$$Desiderability(C++) = \frac{Benefit(C++)}{Cost(C++)} = 23.2$$

$$Desiderability(C) = \frac{Benefit(C)}{Cost(C)} = 10.98$$

$$Desiderability(Python) = \frac{Benefit(Python)}{Cost(Python)} = 36.05$$

# Classifica

---

- Possiamo ordinare le varie alternative in base al grado di desiderabilità calcolato:
  1. Java
  2. Python
  3. C++
  4. C



# Conclusioni

---

- Java rappresenta l'alternativa migliore, in quanto ha raggiunto il valore di desiderabilità maggiore:
  - $Desiderability(Java) = 53.0$
- Python rappresenta una buona alternativa, in quanto ha raggiunto un valore di desiderabilità di poco minore rispetto a Java:
  - $Desiderability(Python) = 36.05$
- C++ rappresenta un'alternativa mediocre, in quanto ha raggiunto un valore di desiderabilità significativamente inferiore rispetto a Java:
  - $Desiderability(C++) = 23.2$
- C rappresenta un'alternativa sconsigliata, in quanto ha raggiunto il più basso livello di desiderabilità tra tutte le alternative:
  - $Desiderability(C) = 10.98$

# Riferimenti

---

- <https://www.solunet.it/cose-alta-affidabilita-informatica-azienda-high-availability/>
- <https://in.springboard.com/blog/programming-language/>
- <https://data-flair.training/blogs/pros-and-cons-of-java/>
- [https://docs.oracle.com/cd/A97335\\_02/ias.102/a95201/avail.htm](https://docs.oracle.com/cd/A97335_02/ias.102/a95201/avail.htm)
- <https://data-flair.training/blogs/advantages-and-disadvantages-of-cpp/>
- <https://data-flair.training/blogs/advantages-and-disadvantages-of-c/>
- <https://www.netguru.com/blog/python-pros-and-cons>
- <https://careerkarma.com/blog/python-java-integration/>
- <https://medium.com/javarevisited/top-5-programming-languages-for-web-development-in-2021-f6fd4f564eb6>