**Immagine che contiene interni

Descrizione generata automaticamente**

**Documentazione Object Design**

**“GLITCH”**

**Progetto presentato da:**

Annunziata Elefante

Ferdinando Napolitano

Santolo Mutone

**Indice**

**1. Introduzione . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3**

* 1. Trade-off . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
  2. Componenti off-the-shelf. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
  3. Linee guida per la documentazione dell’interfaccia. . . . . . . . . . . . . . . . . . . 3
  4. Design pattern . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
  5. Definizioni, acronimi e abbreviazioni . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

**2. Packages**

**3. Interfacce delle classi**

1. **Introduzione**

Con l’Object Design Document specifichiamo in modo dettagliato le decisioni prese in fase di analisi e di design riguardo Glitch; in particolare verranno specificati i principali trade-offs, descritte le componenti off-the-shelfs utilizzate dal sistema, le linee guida per la documentazione delle interfacce e l’individuazione dei Design Patterns. Inoltre, verranno definiti i packages, le class interfaces e i class diagram.

* 1. **Trade-off**
* **Build VS Buy**

Questo trade-off ci mette davanti un quesito molto importante sul futuro del nostro software: costruire VS comprare. Per avere un sistema originale e che rispecchi al meglio i nostri bisogni e le nostre idee, si preferirebbe costruire interamente il sistema; tale scelta porta ad un aumento del lavoro da parte del gruppo che dovrebbe costruire il proprio software consentendo una migliore applicazione sulla realtà di interesse. Tale scelta viene presa in considerazione soprattutto quando non si dispone di un budget elevato. Quando però alcuni fattori vengono a mancare, come ad esempio tempo, tools o un gruppo tecnico, si preferisce comprare il sistema; tale scelta porta ad ottenere dei risultati ottimali che verrebbero consegnati in un tempo più ristretto. Pertanto, il nostro software verrà costruito da zero da parte dell’intero gruppo al fine di ottenere un sistema unico nel suo genere;

* **Security VS Performance**

Ogni sistema dovrebbe offrire in egual misura sia un’ottima sicurezza che alte performance, ma queste non sono direttamente correlate, anzi, guadagnare in una vuol dire perdere nell’altra. A tal proposito è importante raggiungere il giusto equilibrio e prendere in considerazione gli aspetti principali di entrambe per capire quale prediligere. Sebbene avere un sistema ad alte performance vuol dire avere tempi di risposta e compilazione molto bassi e un buon throughput, si è deciso di dare una maggiore importanza alla sicurezza così da garantire la protezione dei dati utente e una giusta gestione dei permessi sulle operazioni e sui servizi;