PROJECT PLAN

*1. Introduzione*

Nel panorama dei mercati finanziari si sta inserendo sempre più la figura dell'analista quantitativo, ovvero colui che utilizza analisi statistiche per studiare i mercati finanziari. L'obbiettivo dei traders è quello di creare strategie algoritmiche, che possano essere testate sui dati del passato tramite un software, così da

poterne valutare le prestazioni. L'obbiettivo del nostro software è quello di fornire ad un analista uno strumento con il quale possa fare le seguenti operazioni:

* ottenere le serie storiche di diversi titoli da più piattaforme (Binance, Yahoo Finance, ...);
* scrivere ed implementare diverse strategie come algoritmi;
* testare le strategie sui dati ed ottimizzare i loro parametri;

E' importante, visti questi tre punti, che il software abbia un'architettura modulare la quale permetta di modificare, in base alle necessità, una delle tre componenti del software; ad esempio la possibilità di implementare una nuova strategia senza modificare la raccolta dati o l'ottimizzazione.

Il software viene curato e prodotto da Francesco Corrini ed Enrico Perani.

*2. Process model*

Il modello utilizzato per la progettazione del software è quello dei prototipi: divideremo lo sviluppo in time slot da una settimana l'uno, in ogni slot si svilupperà un prototipo.

*3. Organizzazione del processo*

Ogni venerdì verrà fatto un briefing nel quale si valuteranno gli sviluppi della settimana passata, si discuteranno/approveranno eventuali proposte di modifica, si scriverà del codice insieme (programmazione a coppie) e si programmeranno le attività per la settimana successiva, le quali saranno divise equamente fra i due membri.

*4. Standard e procedure*

Il progetto sarà scritto in linguaggio java con la convenzione lower Camel case mentre l'ambiente di sviluppo utilizzato sarà Eclipse. Per scrivere la documentazione in linguaggio naturale verranno usati file di testo; per produrre la documentazione sotto forma di diagrammi useremo il linguaggio UML con il software StarUML.

Per gestire le versioni del software nonché lavorare contemporaneamente al progetto e per condividere la documentazione verrà utilizzato GitHub.

*5.Attività management*

Ad ogni briefing il "management" (entrambi perché il progetto è agile) produrrà un documento di sintesi del briefing nel quale saranno riportati:

1. Obiettivi raggiunti e mancati;
2. Stato di avanzamento generale del progetto;
3. Nuove richieste di implementazione;
4. Organizzazione del lavoro nel team.

*6. Rischi*

Il rischio principale è costituito dal fatto che il software prodotto non sia user friendly: per poter utilizzare questo software bisogna essere in grado di implementare le strategie prodotte dagli analisti in codice java per poi aggiungerle nel software. Risulta necessario che l'utente sia un programmatore, o che abbia alle spalle un team di sviluppo (e quindi sostenga un costo) per utilizzarlo. Questo fatto potrebbe scoraggiare molti utenti, che preferirebbero software più commerciali.

*7. Staff*

Francesco Corrini, 21 anni, Studente di Ingegneria Informatica presso l'università degli studi di Bergamo, esperienza di tesi con l'azienda Huawei nello sviluppo di un'applicazione smartwatch per segnalare terremoti.

Enrico Perani, 31 anni, Studenti di Ingegneria Informatica presso l'Università di Bergamo, ha lavorato presso Banca BCC, SAME Deutz-Fahr ed aeroporto “Il Caravaggio”.

*8. Tecniche e metodi*

//

*9. Qualità del software*

//

*10. Work packages*

Francesco Corrini si occuperà prevalentemente del design, implementazione e della coordinazione del gruppo; Perani Enrico dei requisiti del progetto, della documentazione e della gestione delle risorse.

*11. Risorse*

Si hanno a disposizione dalle 50 alle 75 ore lavorative per ogni membro del gruppo, il quale lavorerà prevalentemente dalla postazione presso la sua abitazione (smart working).

*12. Schedule*

Il termine dello sviluppo è previsto per venerdì 7 gennaio.

*13. Cambiamenti*

Il software reagirà solidamente ai cambiamenti necessari per lo sviluppo visto che il modello utilizzato è quello dei prototipi. La gestione delle versioni, come scritto in questo documento, sarà effettuata con GitHub.

*14. Consegna*

La consegna al "cliente" (prof. Angelo Gargantini) sarà effettuata tramite GitHub, a cui avrà accesso e grazie al quale potrà verificare in tempo reale lo sviluppo e le versioni del software.