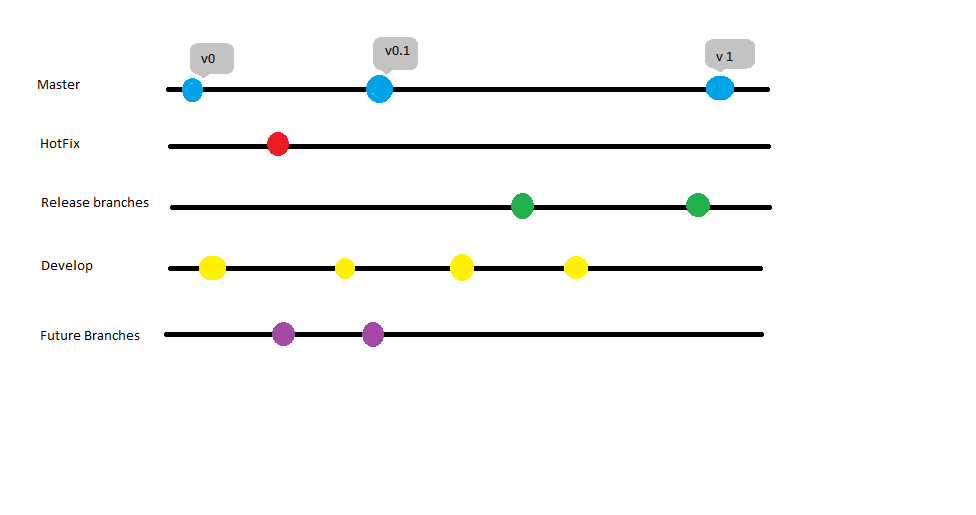
In questo progetto io ed il mio team di sviluppo rilasceremo due versioni, una è quella di prova v0.1 e successivamente quella ufficiale v1.0. Questo progetto prevede vari step che a breve andrò a descrivere. In primo luogo si è previsto una fase di bugfix dove è stato preparato staccandosi dal master. Effettuando un checkout su develop è necessario aggiungere future per il progetto per la riuscita di una buona versione. Dal develop è stata aggiunta una future, creando un nuovo branch con il comando $ git checkout branch future Continuando ad aggiornare questa future e completandola, la mergiamo su develop. Poco prima del merge, il branch Hotfix è stato risolto ed è stato mergiato su master ed è stata fatta una pull da parte del develop richiamando l’Hotfix branch. Effettuati i vari commit su future1 e e mergiandola su develop, il progetto ha bisogno di ulteriori accorgimenti, allora il team ha effettuato una merge da develop al release branch per bugfix di media entità. Da release al develop come detto prima, effettuiamo una pull, per apportare adattamenti conseguenti del bugfix . Il progetto sta per essere rilasciato, prima di mergiarlo al master, effettuiamo un ultimo controllo su release per di bugfix, Infine il progetto viene deployato su master alla versione 1.0.

**Operazioni che non devono essere eseguite per preservare l’integrità del progetto.**

1. Non mergiare una future su master.
2. Iniziare una future nel corso di una release.
3. Non trasportare le modifiche effettuate al branch hotfix e bugfix sul develop.

**Spiegare cosa è una Pull Request e fare la richiesta da GitHub indicando “uiip32” come reviewer.**

Una pull request ($ git request-pull) riceve come parametri il branch di base sul quale vuoi far applicare le modifiche e l'URL del repository Git, essa è una richiesta che viene inviata a tutti i collaboratori.