

# CONTROLUL VERSIUNILOR



+++ cu Git(Hub)

# Cuprins

<b>Glosar</b>	<b>3</b>
<i>Repository (repo)</i>	<b>3</b>
<i>Repo Server</i>	<b>3</b>
<i>Client</i>	<b>3</b>
<i>Working Copy</i>	<b>3</b>
<i>Revision</i>	<b>3</b>
<i>Trunk/Main</i>	<b>3</b>
<i>Head</i>	<b>3</b>
<i>Add</i>	<b>3</b>
<i>Check out</i>	<b>3</b>
<i>Commit</i>	<b>3</b>
<i>Commit Message</i>	<b>3</b>
<i>Changelog</i>	<b>3</b>
<i>Diff</i>	<b>4</b>
<i>Update/Pull</i>	<b>4</b>
<i>Revert</i>	<b>4</b>
<i>Branch</i>	<b>4</b>
<i>Merge</i>	<b>4</b>
<i>Conflict</i>	<b>4</b>
<i>Resolve</i>	<b>4</b>
<b>Introducere</b>	<b>5</b>



# Glosar

## **Repository (repo)**

Locația centrală unde se păstrează versiunile fișierelor.

## **Repo Server**

Serverul care găzduiește un repository.

## **Client**

Calculatorul care se conectează la serverul repo și execută operațiuni gen checkout/add/remove/commit.

## **Working Copy**

Copia locală (pe calculatorul client) a unui proiect, de obicei descărcată de pe repository.

## **Revision**

O versiune a unui document. (v1, v2, v3...).

## **Trunk/Main**

Directorul rădăcină de pe serverul repo.

## **Head**

Cea mai nouă versiune din repository.

## **Add**

Adaugă un fișier în repository, încarcă un fișier în repository.

## **Check out**

Descarcă un fișier/proiect de pe repository.

## **Commit**

Înregistrează o nouă versiune în repository a unui fișier (colecție de fișiere) modificat(ă).

## **Commit Message**

Un mesaj asociat unei acțiuni commit care descrie schimbările făcute în noua versiune salvată.

## **Changelog**

O listă a versiunilor (commit-urilor) unui fișier/proiect de obicei însoțită de mesajele asociate fiecărui commit.

**Diff**

Afișează diferențele dintre două versiuni a unui fișier sau dintre fișierul modificat local (pe working copy) și o versiune de pe repository.

**Update/Pull**

O sincronizare a fișierelor din working copy cu ultima versiune descărcată de pe repository. Update-ul nu "urcă" nimic pe repository, doar aduce ultimele modificări de pe server;

**Revert**

Renunțarea la ultimele modificări (locale) făcute într-un fișier din working copy, și revenirea la ultima versiune aflată în repository sau la o versiune la alegere.

**Branch**

Creează o "copie" a unui fișier/proiect pentru modificări "în paralel" fără a afecta starea actuală a unui proiect.

**Merge**

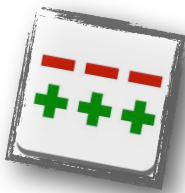
Aplică ultimele modificări dintr-o versiune a unui fișier peste alt fișier (de obicei se întâmplă la "update" când se descarcă ultimele modificări de pe repository).

**Conflict**

Situația în care un "merge" nu se poate executa automat și modificările locale sunt în conflict cu modificările de pe repository.

**Resolve**

Rezolvarea (de obicei manuală) a conflictelor apărute într-un fișier după un merge/update/pull.



# Introducere

Controlul Versiunilor înseamnă gestionarea documentelor (fișierelor) dintr-un proiect, păstrând un istoric al modificărilor făcute atât la nivel de conținut al fișierelor cât și la nivelul structurii proiectului. Modificările sunt înregistrate ca seturi de schimbări între o versiune anterioară și versiunea prezentă sub numele de “revision”-uri. Unei modificări ii se asociază de regulă:

- un identificator (revision number/identifier)
- lista fișierelor modificate (changed files)
- ora la care a fost efectuată înregistrarea modificărilor (commit time)
- persoana care a înregistrat modificările (committer)
- un mesaj care descrie succint modificările făcute

“Revision identifier” este de obicei un număr care crește cu o unitate la fiecare înregistrare a unei modificări (commit).