

# SOFTWARE ENGINEERING LAB ASSIGNMENT 1

BY  
T A SUPRIYA  
16IT142

## **COMPARING VARIOUS VERSION TOOLS**

### **CVN:**

It is installed virtually everywhere.  
The code base isn't as fully featured as SVN.  
It's a very simple system for making sure files and revisions are kept up to date.

### **SVN:**

Very extensively used..Google Code uses Subversion exclusively to distribute code.  
Many different Subversion clients are available.  
Eg: Tortoise SVN for windows, Versions for Mac

### **GIT:**

It is the new fast-rising star of version control systems.  
Git offers a much different type of version control.  
It's a distributed version control system.  
Other version control systems, such as SVN and CVS, use centralized version control.  
It is a fast and efficient system.  
Git isn't as easy to pick up as CVS or SVN, so it's much harder to use for a beginner.

### **Mercurial:**

It was designed for larger projects.  
Small development teams can also use it.  
It is extremely fast, and performance is the most important feature.  
Mercurial is a much simpler system than Git as there aren't as many functions to learn.  
It also comes equipped with a stand-alone Web interface and extensive documentation on understanding Mercurial.

### **Bazaar:**

It offers a very friendly user experience.  
It supports many different types of workflows. Eg: Solo and Centralised.  
It fits to almost any scenario of users and setups.  
It's so easy to modify and also embeddable.

### **Monotone:**

It is the youngest.  
It places higher value on integrity than performance.  
Extensive validation and authentication required.  
Monotone is fairly easy to learn.  
It can import previous CVS projects.