**Models -> PackageVersion -> Constructor**

1. Test the constructor if it sets the appropriate passed values

**Models -> PackageVersion -> Properties**

1. Test for valid and invalid value **Major**
2. Test for valid and invalid value **Minor**
3. Test for valid and invalid value **Patch**
4. Test for valid and invalid value **VersionType**

**Models -> Project -> Constructor**

1. Test the constructor if it sets the appropriate passed values
   * Test if **PackageRepository** is set correctly for **optional** parameter **packages**
   * Test if **PackageRepository** is set correctly for **passed** parameter **packages**

**Models -> Project -> Properties**

1. Test if **Name** is set correctly
2. Test if **Location** is set correctly

**Models -> Package -> Constructor**

1. Test the constructor if it sets the appropriate passed values
   * Test if **Dependencies** is set correctly for **optional** parameter **dependencies**
   * Test if **Dependencies** is set correctly for **passed** parameter **dependencies**

**Models -> Package -> Properties**

1. Test if **Name** is set correctly
2. Test if **Version** is set correctly
3. Test if **Url** is set correctly

**Models -> Package -> CompareTo**

1. Test for valid and invalid value **other**
2. Test if the passed package is not with the same name
3. Test for package passed to be **higher** version
4. Test for package passed to be **lower** version
5. Test for package passed to be **the same** version

**Models -> Package -> Equals**

1. Test for valid and invalid value **other**
2. Test if the passed object is **IPackage**
3. Test for package passed to be **equal** to the package
4. Test for package passed to be **NOT equal** to the package

**Commands -> InstallCommand -> Constructor**

1. Test the constructor if it sets the appropriate passed values

**Commands -> InstallCommand -> Fields**

1. Test for right value set for the **installer //Kato gornoto**
2. Test for right value set for the **package**

**Commands -> InstallCommand -> Properties**

1. Test for right value set for the **Operation** of the installer

**Commands -> InstallCommand -> Execute**

1. Test for calling the perform operation from the installer

**Core -> PackageInstaller -> Constructor**

1. Test for restoring packages when the object is constructed
   * should test when there are dependencies in the project

**Core -> PackageInstaller -> PerformOperation**

**Recursion**

1. Test for **Install** command and empty dependencies list - should call two times **Download** and one time **Remove**

2. Test for **Install** command and **at least one dependency** in the dependencies list - **every dependency on the chain will multiply the calss to the Download and Remove mehtod by 2**

**Repositories -> PackageRepository -> Add**

1. Test for valid and invalid value **package**
2. Test for adding the package when the package does not exist
3. Test for package already exist with the same version
4. Test for calling the Update method when the package exist but with lower version
   * **DO NOT MOCK the System under test**
5. Test for package with **higher** version already exist
6. Test for package with **lower** version already exist

**Repositories -> PackageRepository -> Delete**

1. Test for valid and invalid value **package**
2. Test for package not found
3. Test for package found but is a dependency of other projects and cannot be removed
4. Test for returning the package deleted

**Repositories -> PackageRepository -> Update**

1. Test for valid and invalid value **package**
2. Test for package not found
3. Test for successfully updated package when there is package found with lower version
4. Test for found package with higher version
5. Test for found package with the same version

**Repositories -> PackageRepository -> GetAll**

1. Test for repository with no passed collection to return as parameter empty collection
2. Test for repository with passed collection as parameter to return collection with equal number of elements

**Hints**

**Be careful with the ILogger interface. Take a look at which assembly you are refering**