Web and Mobile app development with Ionic & Angular

Environment Setup

In a nutshell

Here's a detailed guide for setting up your development environment with the listed software and configuration steps:

---

1. **Visual Studio Code (VS Code)**

VS Code is a lightweight yet powerful source code editor, popular among developers for its robust feature set and plugin support.

Installation Steps:

1. Go to the [Visual Studio Code download page](#).
2. Choose the installer for your operating system (Windows, macOS, or Linux).
3. Run the installer and follow the on-screen instructions.
4. During installation, select options to:
   - **Add "Open with Code"** action to the right-click menu (recommended).
   - **Add to PATH** (very important for terminal usage).
5. Once installed, launch VS Code and customize the settings as per your preferences.

Extensions to Install:

- **Prettier** for code formatting.
- **ESLint** for JavaScript linting.
- **GitLens** for enhanced Git capabilities.
- **Debugger for Chrome** for debugging frontend applications.
- **Ionic/Angular Essentials** (for Ionic and Angular projects).

---

2. **Git and Git SCM**

Git is a version control system that tracks changes to files. Git SCM provides a graphical interface for Git.

Installation Steps:

1. Go to the [Git download page](#).
2. Download the installer for your operating system.
3. Run the installer and follow the steps. During the setup:
   - Choose the **default editor** used by Git (select VS Code if installed).
   - Select **Git from the command line and also from third-party software**.
   - Enable **Git Credential Manager** for managing credentials.
   - Configure **line ending conversions** as per your preference.
4. Finish the installation.

Verify Installation:

Open a terminal or command prompt and type:

bash
`git --version`

This should display the installed Git version.

---

3. Generate SSH Key using `ssh-keygen`

SSH keys are used for secure communication with services like GitHub.

Steps to Generate SSH Key:

1. **Open PowerShell** as Administrator.
2. Run the command:
   bash
   `ssh-keygen -t ed25519 -C "your_email@example.com"`
   - Replace `"your_email@example.com"` with your email address used for GitHub.
3. Press **Enter** to accept the default path.
4. When prompted, enter a **passphrase** (optional but recommended for security).
5. This will create two files in the specified directory:
   - `id_ed25519` (private key)
   - `id_ed25519.pub` (public key)

Add SSH Key to GitHub:

1. Copy the content of your **public key** using:
   bash
   `cat ~/.ssh/id_ed25519.pub`
2. Go to your **GitHub account**:
   - Click on your profile picture.
   - Navigate to **Settings** > **SSH and GPG keys**.
3. Click **New SSH Key**.
   - Provide a **title** (e.g., `My Laptop`).
   - Paste the **public key**.
4. Click **Add SSH Key**.

Test SSH Connection:

Run the following command in PowerShell:

bash
`ssh -T git@github.com`

You should see a message saying:

vbnet
`Hi username! You've successfully authenticated, but GitHub does not provide shell access.`

---

4. **Node.js**

Node.js is a runtime environment for executing JavaScript code outside a web browser.

Installation Steps:

1. Go to the Node.js download page.
2. Download the **LTS version** for stability.
3. Run the installer and follow the instructions.
4. Once installed, open **PowerShell** and verify the installation:
   bash
   `node -v`
   `npm -v`
   This should display the versions of Node.js and npm (Node Package Manager).

Manual Installation (using prebuilt binaries):

1. Download the **prebuilt binaries** from the Node.js website.
2. Extract the files to a folder (e.g., `C:\Program Files\nodejs`).
3. Copy the **extracted folder path**.
4. Go to **System Environment Settings**:
   - Right-click **This PC** > **Properties** > **Advanced System Settings**.
   - Click on **Environment Variables**.
   - Find the **Path** variable under **System variables**, and click **Edit**.
   - Add the **Node.js binary path** to the list (e.g., `C:\Program Files\nodejs`).
5. Click **OK** to save and restart your terminal.

Enable Script Execution in PowerShell:

By default, PowerShell restricts the execution of certain scripts.

1. Run PowerShell as **Administrator**.
2. Execute:
   bash
   `Set-ExecutionPolicy Unrestricted`
3. Confirm with **Y** (Yes).

Unblock npm Files:

If you encounter execution issues, unblock npm files with:

bash
`Unblock-File "path_to_npm"`
Replace `path_to_npm` with the path to the npm file in your Node.js directory.

---

5. Install Ionic CLI and Nodemon Globally

Ionic CLI helps in building hybrid mobile apps, and Nodemon is used for automatically restarting the server during development.

Installation Steps:

1. Open PowerShell and run:
   bash

```
npm install -g @ionic/cli nodemon
```

- `@ionic/cli`: The CLI for developing Ionic applications.
- `nodemon`: A utility that monitors for any changes in your source code and automatically restarts your server.

Verify Installation:

Run:

bash
```
ionic --version
nodemon --version
```

This should display the installed versions.

---

6. **MySQL**

MySQL is a popular relational database management system.

Installation Steps:

1. Go to the [MySQL download page](#).
2. Download the **MySQL Installer Community**.
3. Run the installer and choose the **Full** setup type.
4. Follow the on-screen instructions to install:
   - **MySQL Server** (mandatory)
   - **MySQL Workbench** (optional, recommended for GUI management)
   - **MySQL Shell** (optional, for advanced users)

Initial Configuration:

1. Configure the **root** password during installation.
2. Optionally, create a **new user** with limited privileges.
3. Set the **default server** as the **MySQL instance**.

Verify Installation:

1. Open MySQL Workbench or the **MySQL Command Line Client**.
2. Connect using the **root** credentials.
3. Execute:

```
SHOW DATABASES;
```
This should list the system databases.

By following these steps, your development environment will be fully set up and ready for project development. Let me know if you need further assistance or want to add more configurations!