CT Scan Preprocessing for Ischemic Stroke Detection

End User and Technical Guide

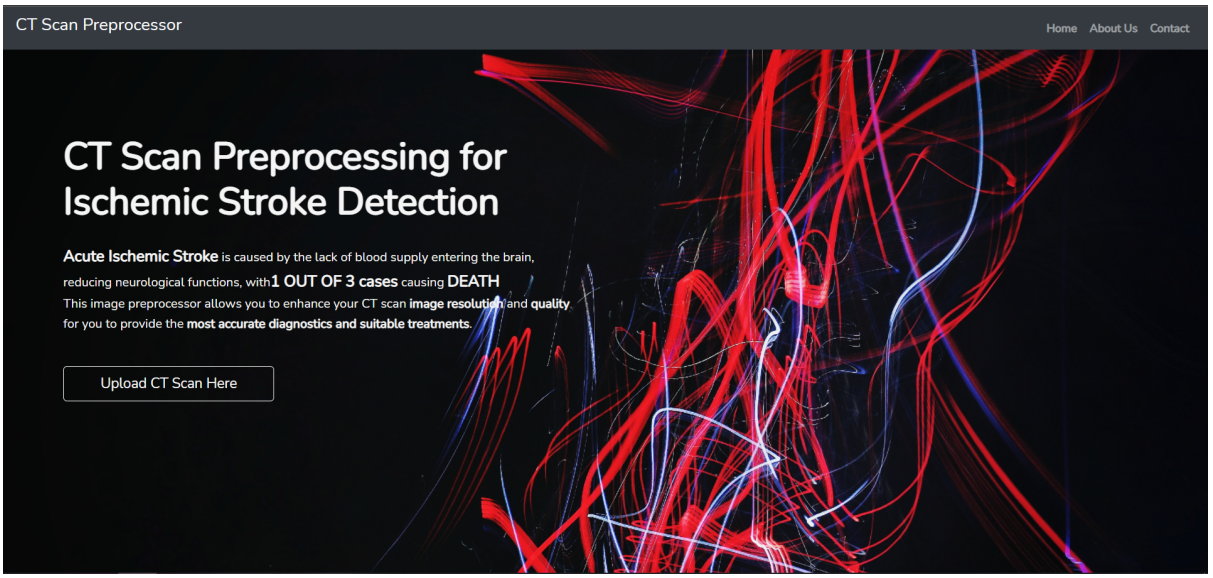
Acute Ischemic Stroke *is caused by the lack of blood supply entering the brain, reducing neurological functions, with* ***1 out of 3 cases*** *causing* ***death****.*

This image preprocessor allows you to enhance your CT scan ***image resolution*** *and* ***quality*** *for you to provide the most* ***accurate diagnostics and suitable treatments****.*

Table of contents

End user guide

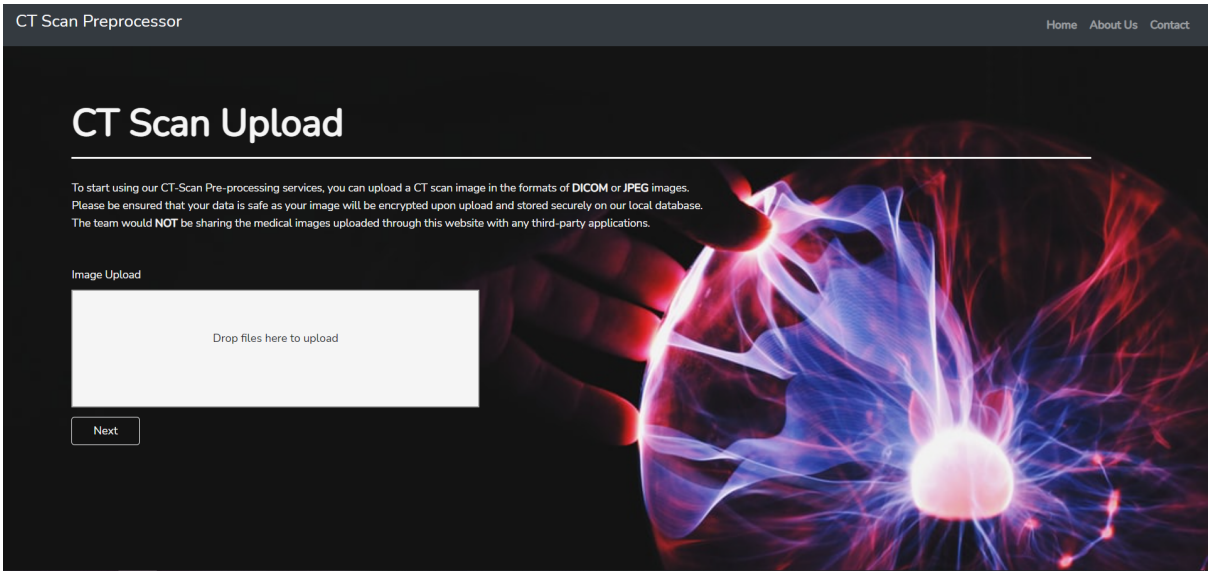
When you first open the web application, you will be greeted with this landing page:



Click **Upload CT Scan Here** to begin.

Note: Most modern browsers should support the web application. If your browser does not support the application, or some functionalities are not working, please consider upgrading or changing your browser.

You will then be brought to the upload page:



There are two ways to upload the images, by clicking on the Dropzone box (the white box with the words ‘Drop files here to upload’) and selecting the files you wish to upload, or you can also drag and drop the files from your computer’s file manager.

Note that you can only upload CT scans in the form of DICOM or JPEG files. It does not matter whether the file extensions are ‘long’ file extensions (i.e. .dicom, .jpeg) or

‘8.3/DOS-styled’ file extensions (i.e. .dcm, .jpg). Also, the total maximum size of files that can be uploaded is 512 MB.

Clicking on the Next page will bring you to the confirmation page, where you can review the images you have selected. You can then proceed to confirm the selected images using the **Submit** button, where the application will analyze the images you have uploaded and display whether it detected a possible stroke, or go back to the image upload page using the **Back** button.

To exit the web application, simply close the browser tab.

Technical guide

Setting up

The following guide helps you to set up the web application on the following operating systems:

- Windows 7 or later
- macOS 10.14 Mojave or later
- Linux (most up-to-date distributions support the application)

If your operating system is not on the list:

- If you are using
    - Windows Vista or earlier, or
    - macOS 10.13 High Sierra or earlier,

  consider upgrading your operating system.
- If you are on another operating system not mentioned above, please contact us for assistance. Depending on your operating system, it may or may not support the application.

The minimum version of PHP that is required for this application is **7.4**. **Composer**, **NPM**, and **MySQL** are also required.

The MySQL database is configured to connect at `localhost:3306`. If you are using a different port, or set a different username/password for MySQL, adjust the settings accordingly in the `.env` file. It is recommended to set a non-root username and password as this is more secure.

It can be helpful to connect to a frontend GUI to manage the database. You can use applications like MySQL Workbench, LibreOffice Base (should be pre-installed in many Linux distributions or otherwise available in your package manager), or, if you have a suitable Microsoft 365 subscription plan, you can use Microsoft Access (only for Windows users as well as requiring an ODBC driver).

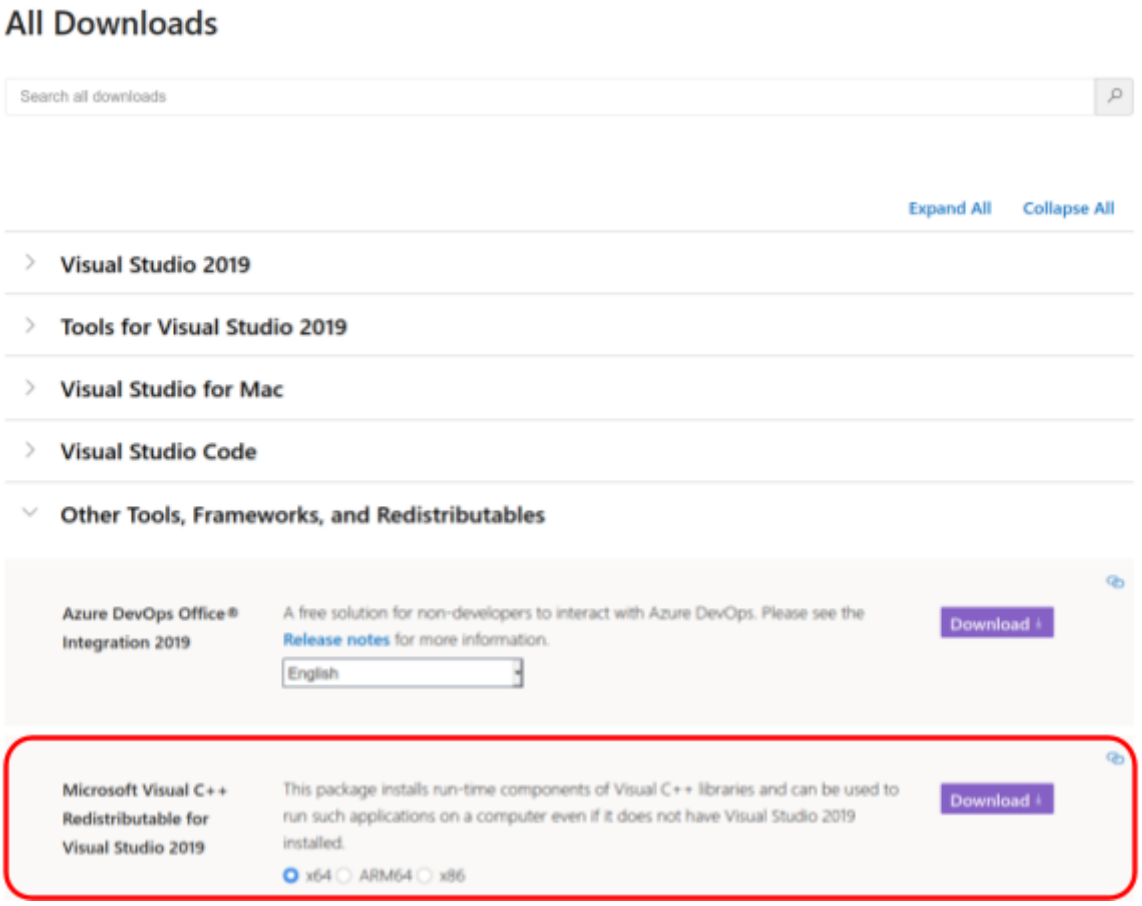
Python (version 3.7 or above) is also required for the back end.

Windows

If you are planning to install the software in Windows Subsystem for Linux (WSL), refer to either the Linux or macOS installation instructions in the following section.

Make sure that you already have the **Microsoft Visual C++ Redistributable for Visual Studio 2019** downloaded; you can check whether you have downloaded it or not from the Programs and Features section of the Control Panel or the Windows 10 Settings application. Otherwise, download it at https://visualstudio.microsoft.com/downloads/. **You only need to download the runtime library, NOT the entire Visual Studio IDE**.

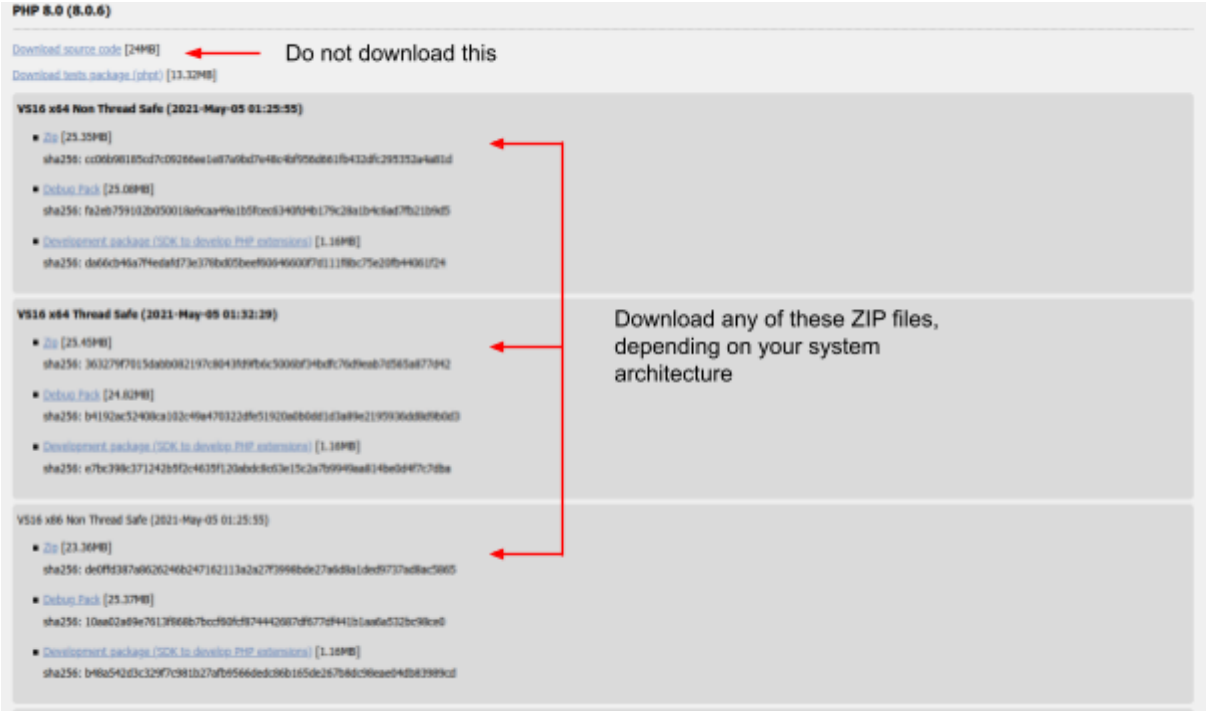
On the Visual Studio download page, scroll all the way down to the bottom of the page to the **All Downloads** section, then under the **Other Tools, Frameworks and Redistributables** sub-section, download the **Microsoft Visual C++ Redistributable for Visual Studio 2019** executable:



Choose the installer that is appropriate for your processor architecture; for example, if you are using a 64-bit system, download the x64 installer. If you are using a 32-bit system, use the x86 installer.

The redistributable is required to run PHP on your system. Download the appropriate version of PHP from https://windows.php.net/download/. Remember that it has to be version 7.4 or later. Choose the version that is suitable with the architecture of your system (i.e. 32-bit (x86) installation for 32-bit systems, and 64-bit installation for 64 bit systems). Download the zipped folder containing the **pre-compiled binaries** (**not** the source code) and extract it to a

location of your choice.



Before we install Composer, update your PATH variable so that you can call php in the terminal. To do this, press ⊞ Win + R to bring up the Run dialog box, type in **`sysdm.cpl`**, then hit ↵ Enter. A **System Properties** dialog box should appear. Go to the **Advanced** tab, and click on **Environment Variables**. There will be two PATHs, one under the category **User variables for %USERNAME%** (where %USERNAME% is your Windows user account name), and another under **System variables**.

- If you want to install PHP for all users (recommended), update the PATH under **System variables**.
- Otherwise, if you want to use PHP for your own user account, update the PATH under **User variables for %USERNAME%.**

Double-click on one of the PATH variables above, then click **New** in the dialog box that appears. Enter the location of the folder you extracted PHP earlier. For example, if you install PHP in `C:\path\to\your\php\php-8.0.3-Win32-vs16-x64`, then add this location to the PATH. Click **OK** to dismiss all dialog boxes.

Open a new terminal window (Command Prompt, PowerShell or Bash will work). If you already have a terminal window open prior to setting your PATH, close it and relaunch the terminal. If you are using Windows Terminal, closing all active tabs and opening a new one will also work. Run this command to verify the installation:

```
php --version
```

If you set your PATH correctly, it will print the PHP version that you just installed.

Download the Composer installer at https://getcomposer.org/Composer-Setup.exe and follow the instructions on-screen. Use the recommended settings in the installer.

Download Node.js at https://nodejs.org/. Again, run the installer and follow the instructions on-screen. This will automatically install `npm`.

The Composer and Node.js installers will set the PATHs of `composer` and `npm` by default.

Follow the instructions at https://dev.mysql.com/doc/refman/8.0/en/windows-installation.html to download and install MySQL.

Download and install Python from https://docs.python.org.

Installing with Homebrew on macOS, Linux or WSL

If you are using macOS or Linux (including Windows under WSL), you can choose to install the necessary software with **Homebrew**, a free package manager. Install Homebrew on macOS by going to the official website at https://brew.sh, and follow the instructions. If you are using Linux or WSL, refer to https://docs.brew.sh/Homebrew-on-Linux instead.

Run the following commands in your terminal after installing Homebrew:

```
brew install php
brew install composer
brew install nodejs
brew install mysql
```

Installing MySQL through Homebrew allows the use of the root account without a password. If you have set a password during the installation, or use any account other than root, remember to modify the .env file.

For macOS users, Python can also be downloaded from the official website. For Linux users, Python usually comes pre-installed on the system. If the Python installation is not Version 3.7 or above, or your Linux system does not come with Python, install Python with your preferred package manager.

Using another package manager on Linux/WSL

Alternatively, you may choose to use another package manager (e.g. apt-get or yum) on a Linux distribution or an installation of WSL. Please refer to the installation instructions for your distribution as different distributions will have different installation instructions.

You may choose to download certain packages with Homebrew, and others with another package manager. For example, you may choose to download PHP, Composer and NodeJS with apt-get, and MySQL with Homebrew. This is useful because versions of MySQL downloaded through apt-get require elevated privileges to login as root, which may cause complications in the database connection unless you plan to use another MySQL account.

Database setup

Start the MySQL server and log in. Create a database called `web_app` with the following SQL command:

```
CREATE DATABASE web_app;
```

Build and install

There are three install scripts provided. Depending on which terminal you use, execute one of the following:

- **`INSTALL.sh`** if you are using a Unix shell (including Bash or Zsh),
- **`INSTALL.bat`** if you are using the Windows Command Prompt, or
- **`INSTALL.ps1`** if you are using PowerShell.

Run

To start the web application, run the following command:

```
php artisan serve
```

By default, the application will run at port 8000. If you want to use another port (usually because it is already in use for something else), specify the port using the `--port` flag (for example, the following runs the application at port 8001:

```
php artisan serve --port=8001
```