

Documentation for the Task Management
System, EULA and User Manual

Task Management System Documentation

Authour: S275931

Task Management System

Documentation

Table of Contents

Introduction	3
End User License Agreement	3
Terms of Use	4
System Requirements	5
Docker Deployment (Windows):.....	5
Docker Deployment (macOS):.....	6
Docker Deployment (Linux):.....	6
Render.com deployment (online):.....	7
Building and deploying from source code (Windows):.....	7
Building and deploying from source code (macOS):	8
Building and deploying from source code (Linux):	8
Installation	9
Setup Render.com Deployment	11
Installation and Setup using Docker Deployment	14
Installation and Setup Using Source Code.....	22
Prerequisites	23
Downloading from the GitHub repository.....	23
Open the PyCharm Project.....	24
Setting up the PyCharm environment (Windows).....	27
Setting up the PyCharm Environment (Linux)	34
Setting up the PyCharm environment (macOS).....	38
Setting up the Flask Configuration	43
Run the Task Management System from PyCharm	46
Setting up and pushing to Git repository (Version Control).....	48
Modifying Source code, Adding and Testing New Features.....	55
Suggested Modifications.....	56
Unit Testing the Task Management System Source Code.....	58
Deploy online using Render.com or other webhosting service using a Continuous Integration / Continuous Deployment (CI/CD) Pipeline.	61
Deploy using Docker or other containerisation platforms.....	71
Using the Task Management System	77

First Time Setup	77
Logging out	80
Logging In.....	81
Create Account.....	82
Manage Projects	84
Create a New Project	84
View Projects	85
Update a Project – Administrators and Root User Only	87
Delete a Project – Administrators and Root User Only.....	88
Manage Tasks.....	90
Create a New Task.....	90
View Tasks.....	91
Update Tasks – Administrators and Root User Only	92
Delete Tasks – Administrators and Root users only.	94
Managing Your Account	95
View Your Account Details	95
Changing your Account Details	96
Changing your Username.....	98
Changing your Password	99
Deleting your Account.....	101
Managing Users and Administrative Tasks – Administrators and Root User Only	103
View the List of Users as a Table (Administrators and Root User Only).....	104
Change User Account Type (Administrators and Root Users Only)	105
Delete a User (Administrators and Root Users Only).	107
Blocking a User (Administrators and Root Users Only).....	108
Unblocking a User (Administrators and Root Users Only)	110
Changing Another User’s Password (Administrators and Root Users Only)	112
Factory Resetting the Task Management System (Administrators and Root Users Only)	114
Task Management System on Mobile Devices.	116
Security Features	130
References	131

Introduction

The Task Management System is a Python-based web application that manages tasks and projects for an organization to help the organisation keep track of tasks and their associated projects. Features include a database that contains three tables (one with a list of tasks, another with a list of projects and thirdly with a list of users), user roles (Administrator, Root User, Standard User), Create, Read, Update and Delete functionality for tasks, projects, and users. Administrators have full control of the Task Management system (Add, Update, View and Delete tasks and projects, manage other users (change account type and password, view details of all users, delete users, block, and unblock users), factory reset the Task Management System, and manage their own account. Standard users are restricted therefore can only add and view tasks and projects, view and manage their own account. Standard users cannot update or delete tasks and projects, view and manage other users or factory reset the Task Management system. For security, authentication is required for all users and all accounts are password protected. The passwords are encrypted and stored as hashes in the database in the event the application or database gets hacked, randomly generated session IDs authenticate each session, Passwords are required to have minimum length of 10 characters and must contain at least one uppercase letter, one lowercase letter, one number, one special character and must not be a common password. The Task Management System is cross-platform and can run as a container on a computer `or server running Windows, Mac OS or Linux with Docker installed, hosted online as a web service on hosting sites such as Render.com linked to a GitHub repository (It could also be hosted on other web hosting platforms that are compatible with Python 3.12, Flask and Gunicorn). You can also deploy the Task Management System directly from its source code by cloning its Git repository, run app.py in a suitable Python IDE such as PyCharm and have the Python 3.12 interpreter installed on your computer, then deploy on a containerisation platform of your choice such as Docker (others may require the Dockerfile to be replaced with their own containerisation file) or on a web hosting platform such as Render.com.

End User License Agreement

IMPORTANT – YOU MUST AGREE TO THE TASK MANAGEMENT SYSTEM’S GNU AGPLv3 LICENSE BEFORE YOU CAN USE THIS PRODUCT IN ANY WAY.

The full license agreement is available in a separate document. The Task Management System is open source with the copyleft GNU Affero General Public License.

You must comply with the following permissions, conditions and limitations when using, managing, downloading, installing, setting up, distributing, deploying, or modifying the Task Management System as stated in the table below in accordance with the open-source GNU Affero General Public License Agreement. Failure to comply can result in legal action under U.K. law, international law, and local laws of your country where you are using this software.

The GNU Affero General Public License Agreement is available here:

<https://www.gnu.org/licenses/agpl-3.0.en.html>

Alternative links to license agreement: <https://github.com/AlexanderJohnRobertson/task-management-system/wiki/Task-Management-System-GNU-AGPLv3-License-Agreement>
[https://github.com/AlexanderJohnRobertson/task-management-system-clean-install/wiki/Task-Management-System-\(Clean-Install\)-Documentation](https://github.com/AlexanderJohnRobertson/task-management-system-clean-install/wiki/Task-Management-System-(Clean-Install)-Documentation)

Permissions	Conditions	Limitations
Commercial Use is Authorised	You Must Disclose the Original Source	Liability – We cannot be held liable for any damage or data loss caused by modifications.
Distribution is Authorised	You Must Include License and Copyright Notice	Warranty – Modified code is not covered by warranty.
Modification is Authorised	Network use is Distribution	
Patent Use is Authorised	You Must Use the Same License	
Private Use is Authorised	You Must State Changes (Choose an open source license, 2007)	

Terms of Use

IMPORTANT – YOU MUST AGREE TO THE TERMS OF USE BEFORE YOU CAN USE THE TASK MANAGEMENT SYSTEM IN ANY WAY.

1. You must accept the license agreement (see above).
2. You must comply with the license agreement and must not violate it or else you could face legal action under U.K. law, international law, and local laws in the country you are using this software.
3. You must not violate the terms of use or else you could face legal action under U.K. law, international law, and local laws in the country you are using this software.
4. You must not use the Task Management System for illegal purposes or to support criminal activity.
5. You must not modify and redistribute the Task Management System by inserting malicious code in any language into the source code to turn it into malware. This includes but not limited to:
 - a. Viruses
 - b. Trojans
 - c. Worms
 - d. Adware and Malvertising
 - e. Spyware
 - f. Ransomware
 - g. Rootkits
 - h. Browser Hijackers
 - i. Potentially Unwanted Programs
 - j. Cryptominers
 - k. Keyloggers
 - l. Password Stealers

- m. Hack tools.
 - n. Browser Hijackers
 - o. Links to third party malware websites, scam websites and spam websites
 - p. Spambots, Bots and Botnets
 - q. Exploits
 - r. Backdoors
 - s. Logic Bombs
 - t. (Distributed) Denial of Service Attack
 - u. Fileless Malware
6. You must comply the relevant laws and regulations of the country(s) you are using the Task Management System in and not break them. Failure to comply can result in civil and criminal prosecution. These include:
- a. Data Protection Act and General Data Protection Regulations
 - b. Computer Misuse Act
 - c. Communications Act and Malicious Communications Act
 - d. Telecommunications Act.
7. You must not use the Task Management System or modify it in a way to facilitate cybercrimes. This includes but is not limited to:
- a. Malware (see above for examples)
 - b. Break the laws listed above.
 - c. Hacking
 - d. Spying and espionage
 - e. Identity Theft
 - f. Any type of fraud
 - g. Phishing
 - h. Spam
 - i. Scams
8. You must not redistribute this software with a proprietary license and sell it for money and pretend it's your own or with any other type of license. The license must stay the same.
9. As the software is currently free, you must not sell it for any amount of money.

System Requirements

Docker Deployment (Windows):

- Computer must meet the system requirements to run Docker and a compatible web browser.
- 64 Bit Windows 10 Home, Professional, Enterprise or Education 21H2 (build 19044) or later or Windows 11 Home, Professional, Enterprise or Education version 22H2 or later (Docker, Inc., 2024).
- Docker Desktop and compatible web browser installed.
- Windows Subsystem for Linux (WSL) 2 feature enabled with a Linux operating system installed (Available from the Microsoft Store) or Hyper-V and Windows Containers features enabled (Docker, Inc., 2024).
- Professional and Enterprise editions of Windows are required to run Windows containers (Docker, Inc., 2024).

- 64-bit processor (CPU) with Second Level Address Translation (SLAT) (Docker, Inc., 2024) (Extended Page Table (EPT) in Intel processors, Rapid Virtualisation Indexing (RVI) in AMD processors) (Gibb, 2011).
- 4GB system RAM (Docker, Inc., 2024) (More may be required if application is scaled).
- Hardware Virtualisation enabled in computer BIOS (Docker, Inc., 2024).
- Internet connection.
- Tested web browsers include Mozilla Firefox, Microsoft Edge, Google Chrome, and Safari but should run on other web browsers too.
- 152 MB disk space for Task Management System container (may increase if database is large)
- 100 GB disk space for Docker as Docker can take up large amounts of disk space (Kinda Code, 2022).
- Mozilla Firefox System Requirements: 1GHz processor (CPU), 2GB RAM, 500 MB disk space (Mozilla, 2024).
- Microsoft Edge is bundled with Windows 10 and 11 (Microsoft, 2024).
- Google Chrome System Requirements: Windows 10 or 11, Intel Pentium 4 processor (CPU) or AMD equivalent (Google, 2024).

Docker Deployment (macOS):

- Docker is compatible with most recent versions of macOS (Docker, Inc., 2024).
- Docker is compatible with Apple Silicon chips (CPUs) such as M1, M1 Pro, M1 Max, M1 Ultra, M2, M2 Pro, M2 Max, M2 Ultra, M3, M3 Pro, M3 Max and future Apple Silicon processors (Docker, Inc., 2024).
- Docker is compatible with most recent Intel CPUs in Macs with Intel chips (Docker, Inc., 2024).
- 4 GB RAM (Docker, Inc., 2024) (more RAM may be required if scaled)
- 100 GB disk space for Docker as Docker can take up large amounts of disk space (Kinda Code, 2022).
- 152 MB disk space for Task Management System container (may increase if database is large)
- Internet Connection.
- Tested web browsers include Mozilla Firefox, Microsoft Edge, Google Chrome, and Safari but should run on other web browsers too.
- Mozilla Firefox System Requirements: macOS 10.15 or later (Catalina), Intel or Apple Silicon processor (CPU), 512 MB RAM, 200 MB disk space (Mozilla, 2024).
- Google Chrome System Requirements: macOS Catalina 10.15 and up (Google, 2024).
- Safari System Requirements: Safari is bundled with macOS (Apple, 2024).

Docker Deployment (Linux):

- Supported Linux Distributions: Ubuntu, Debian, Fedora (Docker, Inc., 2024).
- 64-bit kernel and processor (CPU) for virtualisation (Docker, Inc., 2024).
- KVM virtualization support (Docker, Inc., 2024).
- QEMU 5.2 or later (Docker, Inc., 2024).
- systemd init system (Docker, Inc., 2024).
- Gnome, KDE, or MATE Desktop environment (Docker, Inc., 2024).
- 4 GB RAM (Docker, Inc., 2024) (more RAM may be required if application is scaled).
- Configuring ID mapping in user namespaces enabled (Docker, Inc., 2024).
- Initialize pass recommended for credentials management (Docker, Inc., 2024).

- 100 GB disk space for Docker as Docker can take up large amounts of disk space (Kinda Code, 2022).
- 152 MB disk space for Task Management System container (may increase if database is large)
- Internet Connection.
- Tested web browsers include Mozilla Firefox, Microsoft Edge, Google Chrome, and Safari but should run on other web browsers too.
- Mozilla Firefox System Requirements: Following libraries and packages must be installed: glibc 2.17 or higher, GTK+ 3.14 or higher, libglib 2.42 or higher, libstdc++ 4.8.1 or higher, X.Org 1.0 or higher (1.7 or higher is recommended), DBus 1.0 or higher, NetworkManager 0.7 or higher, PulseAudio. Mozilla Firefox is bundled with some Linux distributions (Mozilla, 2024).
- Google Chrome System Requirements: 64-bit Ubuntu 18.04+, Debian 10+, openSUSE 15.2+, or Fedora Linux 32+, SSE3 capable Intel Pentium 4 or later processor (CPU) or AMD equivalent (Google, 2024).

[Render.com deployment \(online\):](#)

- Any device with compatible browser, operating system, and internet connection.
- Examples of devices include workstations, desktop and laptop computers running Windows, Linux, macOS or ChromeOS, Smartphones running Android and iOS, Tablets and iPads running Android, iOS and iPadOS, Macs running macOS, Chromebooks running ChromeOS, Raspberry Pi etc.
- Tested browsers include Mozilla Firefox, Microsoft Edge, Safari, and Google Chrome. Task Management System should be accessible on other browsers too.

[Building and deploying from source code \(Windows\):](#)

- Internet connection to clone git repository.
- Software: Windows 10 1809 or later or Windows 11, Python 3.12 Interpreter, Python IDE (PyCharm Recommended), Git (to commit to your own git repository (required for Render.com deployments), Docker (for containerisation deployments), GitHub Desktop SQLite Studio. You could try other Python IDEs, containerisation platforms and git clients but may require modifications to the code, Dockerfile etc. and may result in compatibility issues as they have not been tested.
- PyCharm System Requirements (Jetbrains, 2024):
 - 4 GB RAM (8 GB RAM recommended).
 - Any modern processor (CPU) – Multi-core CPU recommended.
 - 3.5 GB Disk Space (5 GB Disk Space on Solid State Drive (SSD) recommended).
 - 1024x768 screen resolution (1920x1080 or higher recommended).
- Docker System Requirements: See above.
- Web Browser System Requirements: See above.
- GitHub Desktop and Git System Requirements: Windows 10 64-bit or Windows 11 (GitHub, 2024).
- GitHub account (GitHub, 2024).
- Docker Hub account (Docker, Inc., 2024).
- Render.com account (Render.com, 2023).
- JetBrains account (Jetbrains, n.d.).
- PyCharm must have Flask and Gunicorn installed using PIP Install.

Building and deploying from source code (macOS):

- Internet connection to clone git repository and deployment.
- Software: macOS 12.0 (Monterey) or later, Python 3.12 Interpreter, Python IDE (PyCharm Recommended), Git (to commit to your own git repository (required for Render.com deployments), Docker (for containerisation deployments), GitHub Desktop SQLite Studio. You could try other Python IDEs, containerisation platforms and git clients but may require modifications to the code, Dockerfile etc. and may result in compatibility issues as they have not been tested.
- PyCharm System Requirements:
 - 4 GB RAM (8 GB RAM recommended).
 - Any modern processor (CPU) – Multi-core CPU recommended.
 - 3.5 GB Disk Space (5 GB Disk Space on Solid State Drive (SSD) recommended).
 - 1024x768 screen resolution (1920x1080 or higher recommended).
- Docker System Requirements: See above.
- Web Browser System Requirements: See above.
- GitHub Desktop and Git System Requirements: macOS 10.13 (High Sierra) or later.
- GitHub account.
- Docker Hub account.
- Render.com account.
- JetBrains account.
- PyCharm must have Flask and Gunicorn installed using PIP Install.

Building and deploying from source code (Linux):

- Internet connection to clone git repository and deployment.
- Software Any Linux distribution that supports KDE or Gnome DE (e.g. Ubuntu, Debian, RHEL etc.), Linux Distribution must include GLIBC 2.27 or later, Python 3.12 Interpreter, Python IDE (PyCharm Recommended), Git (to commit to your own git repository (required for Render.com deployments), Docker (for containerisation deployments), GitHub Desktop, SQLite Studio. You could try other Python IDEs, containerisation platforms and git clients but may require modifications to the code, Dockerfile etc. and may result in compatibility issues as they have not been tested.
- PyCharm System Requirements:
 - 4 GB RAM (8 GB RAM recommended).
 - Any modern processor (CPU) – Multi-core CPU recommended.
 - 3.5 GB Disk Space (5 GB Disk Space on Solid State Drive (SSD) recommended).
 - 1024x768 screen resolution (1920x1080 or higher recommended).
- Docker System Requirements: See above.
- Web Browser System Requirements: See above.
- GitHub Desktop is not supported on Linux. Git is supported.
- GitHub account.
- Docker Hub account.
- Render.com account.
- JetBrains account.
- PyCharm must have Flask and Gunicorn installed using PIP Install.

Installation

This section provides Installation instructions for the Task Management System.

Before installing the Task Management System, the prerequisite software must be installed first:

- For Docker installation and setup instructions, visit the official Docker website:
 - Install Docker on Windows: <https://docs.docker.com/desktop/install/windows-install/>
 - Install Docker on macOS: <https://docs.docker.com/desktop/install/mac-install/>
 - Install Docker on Linux: <https://docs.docker.com/desktop/install/linux-install/>,
<https://docs.docker.com/desktop/install/debian/>,
<https://docs.docker.com/desktop/install/fedora/>,
<https://docs.docker.com/desktop/install/ubuntu/>,
<https://docs.docker.com/desktop/install/archlinux/>.
 - Full Docker Documentation: <https://docs.docker.com/manuals/>
 - Download Docker: <https://www.docker.com/products/docker-desktop/>
- For PyCharm installation and setup, visit the official JetBrains PyCharm website:
 - Installation Guide: <https://www.jetbrains.com/help/pycharm/installation-guide.html>
 - Full PyCharm Documentation: <https://www.jetbrains.com/help/pycharm/getting-started.html>
- Use PIP to install Flask and Gunicorn if they have not already been installed.
- For Flask install and setup instructions and documentation, visit the official Flask website:
<https://flask.palletsprojects.com/en/3.0.x/>
- For Gunicorn install and setup instructions and documentation, visit the official Gunicorn website: <https://docs.gunicorn.org/en/stable/>
- For Python 3.12 interpreter installation and setup instructions and documentation, visit the official Python website:
 - Python 3.12 documentation: <https://docs.python.org/3/index.html>
 - Python Installation Instructions: <https://python.land/installing-python>
 - Python Wiki: <https://wiki.python.org/moin/>
 - Python 3.12 download links: <https://www.python.org/downloads/release/python-3120/> and <https://www.python.org/downloads/>
 - Python Tutorials: <https://www.learnpython.org/>
- For Git download, installation and setup instructions and documentation, visit the official Git website:
 - Git documentation and setup instructions: <https://git-scm.com/doc>
 - Git download link: <https://git-scm.com/downloads>
- For GitHub Desktop installation and setup instructions and documentation, visit the official GitHub Desktop website:
 - GitHub Desktop download link: <https://desktop.github.com/>
 - GitHub Desktop documentation: <https://docs.github.com/en/desktop>
- For GitHub Documentation and instruction manuals, visit the official GitHub website:
 - Documentation and manuals: <https://docs.github.com/en>
 - GitHub website: <https://github.com/>
- For Render.com instructions, manuals and documentation visit the Render.com official website:
 - Documentation: [Docs + Quickstarts | Render Docs](#)
 - Render.com: <https://render.com/>
- For SQLite Studio and SQLite Download and documentation please visit the official SQLite Studio and SQLite websites:
 - SQLite Studio download: <https://sqlitestudio.pl/>

- SQLite Documentation: <https://www.sqlite.org/docs.html>
 - SQLite Tutorials: <https://www.sqlitetutorial.net/>
- For the HTML documentation and manuals, please visit the official HTML website: <https://html.spec.whatwg.org/>
- For the CSS documentation and manuals, visit the official CSS website: <https://www.w3.org/TR/CSS/#css>
- For CSS tutorials, visit <https://www.w3schools.com/Css/>
- For the JavaScript documentation, manuals, and tutorials, visit the official JavaScript website and tutorial websites:
 - Official websites: <https://www.javascript.com/> and <https://ecma-international.org/publications-and-standards/standards/ecma-262/>
 - Documentation: <https://devdocs.io/javascript/>
 - JavaScript Tutorials: <https://www.w3schools.com/js/DEFAULT.asp>
- For Mozilla Firefox Download and documentation, visit the official Mozilla Firefox website:
 - Documentation: <https://developer.mozilla.org/en-US/docs/Mozilla/Firefox> and <https://support.mozilla.org/en-US/products/firefox/get-started>
 - Download: <https://www.mozilla.org/en-GB/firefox/new/>
- For Google Chrome download and documentation, visit the official Google Chrome website:
 - Documentation: <https://developer.chrome.com/docs/>
 - Help Manual: <https://support.google.com/chrome/?hl=en#topic=7439538>
 - Download: <https://www.google.co.uk/chrome/>
- For Microsoft Edge Documentation and download, visit the Microsoft Edge official webpage:
 - Documentation: <https://learn.microsoft.com/en-us/microsoft-edge/>
 - Help: <https://support.microsoft.com/en-us/microsoft-edge>
 - Download: <https://www.microsoft.com/en-us/edge/download?form=MA13FJ>
- For Safari Download and Documentation, visit the official Safari Website:
 - Documentation: <https://developer.apple.com/documentation/safari-release-notes>
 - Support: <https://support.apple.com/safari>
 - Download: <https://support.apple.com/downloads/safari>
- For the Windows Subsystem for Linux documentation, visit the WSL official website:
 - Documentation: <https://learn.microsoft.com/en-us/windows/wsl/>
- For the Hyper-V manuals, visit the official Microsoft Website:
 - Windows 10: <https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/>
 - Windows 11: <https://techcommunity.microsoft.com/t5/educator-developer-blog/step-by-step-enabling-hyper-v-for-use-on-windows-11/ba-p/3745905>
- For download, manuals, and documentation on Windows, visit the official Microsoft Windows website:
 - Documentation: <https://learn.microsoft.com/en-us/windows/>
 - Windows 11: [https://support.microsoft.com/en-us/windows/meetwindows-11-the-basics-a7519756-6807-41e4-be66-ed3b2c0abe0d](https://support.microsoft.com/en-us/windows/meet-windows-11-the-basics-a7519756-6807-41e4-be66-ed3b2c0abe0d)
 - Windows 11 Support: <https://support.microsoft.com/en-us/meetwindows11>
 - Windows 11 Download: <https://www.microsoft.com/en-gb/software-download/windows11/>
 - Windows 10: <https://support.microsoft.com/en-us/microsoft-edge/quick-start-guides-for-windows-10-surface-book-and-microsoft-edge-4e603411-16ad-73f7-0923-5aa3d327bb59>
 - Windows 10 Download: <https://www.microsoft.com/en-gb/software-download/windows10>
- For macOS documentation and manuals, please visit the official Apple macOS website:
 - User Guide: <https://support.apple.com/en-gb/guide/mac-help/welcome/mac>
 - Manuals: <https://support.apple.com/manuals/macos>

- Download (Requires a Mac): <https://support.apple.com/en-us/102662>
- For Linux documentations and downloads, visit the official pages of the Linux distributions:
 - Linux Kernel Documentation: <https://www.kernel.org/doc/html/latest/>
 - Ubuntu documentation: <https://docs.ubuntu.com/>
 - Ubuntu Download: <https://ubuntu.com/download>
 - Debian Documentation: <https://www.debian.org/doc/>
 - Debian Download: <https://www.debian.org/distrib/>
 - Fedora Documentation: <https://docs.fedoraproject.org/en-US/docs/>
 - Fedora Download: <https://www.fedoraproject.org/en/workstation/download> and <https://docs.fedoraproject.org/en-US/fedora/latest/fedora-downloads-info/>
 - Arch Linux Documentation: <https://wiki.archlinux.org/> and https://wiki.archlinux.org/title/Installation_guide
 - Arch Linux Download: <https://archlinux.org/download/>

Setup Render.com Deployment

To set up the Render.com deployment, you just need a device that connects to the internet and a recent version of a web browser installed.

The clean installation of the Task Management System is available on Render.com here: <https://task-management-system-clean-install.onrender.com/setup>.

Another version of the Task Management System with example data, users, tasks, and projects is available here: <https://task-management-system-hnmb.onrender.com/>

The username for the Root user is Root and the password is TaskManagement123#

The username for an administrator is barneyPurpleDinosaur and the password is BabyBop123#

The username for a standard user is Birmingham888 and the password is SouthSide123@

Login Credentials for Task Management System With Example Data

User Role	Username	Password
Root User	Root	TaskManagement123#
Administrator	barneyPurpleDinosaur	BabyBop123#
Standard	Birmingham888	SouthSide123@

When you click the first link, you will be redirected to the setup page.

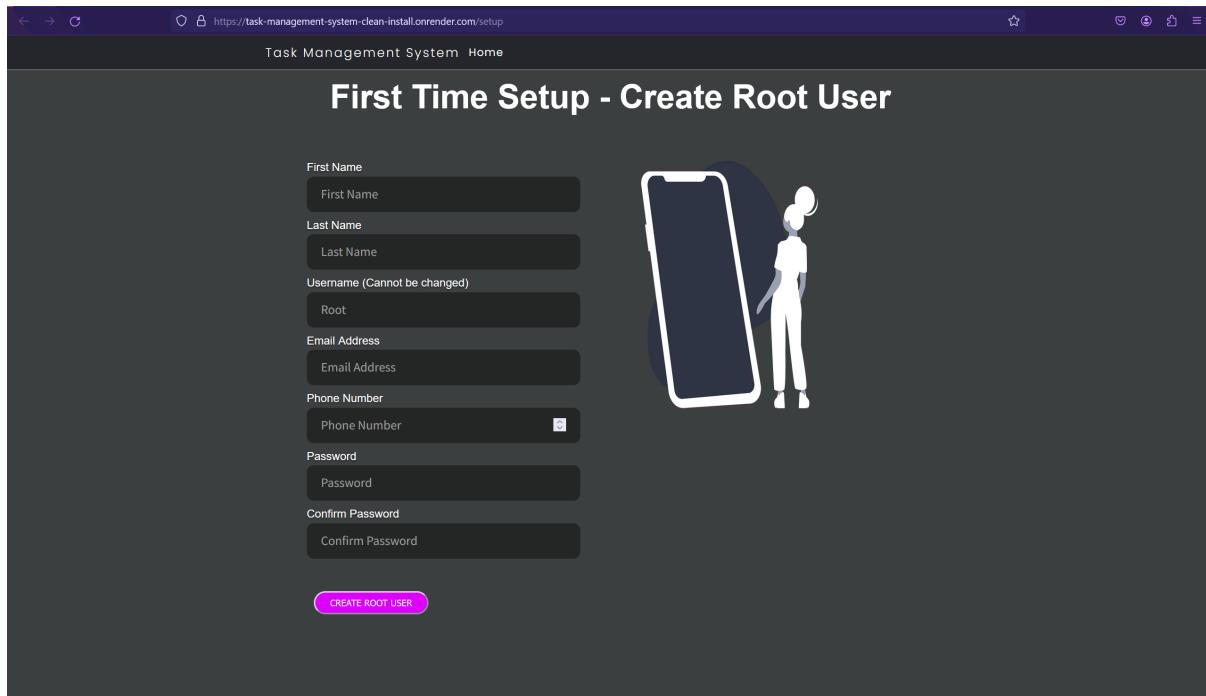
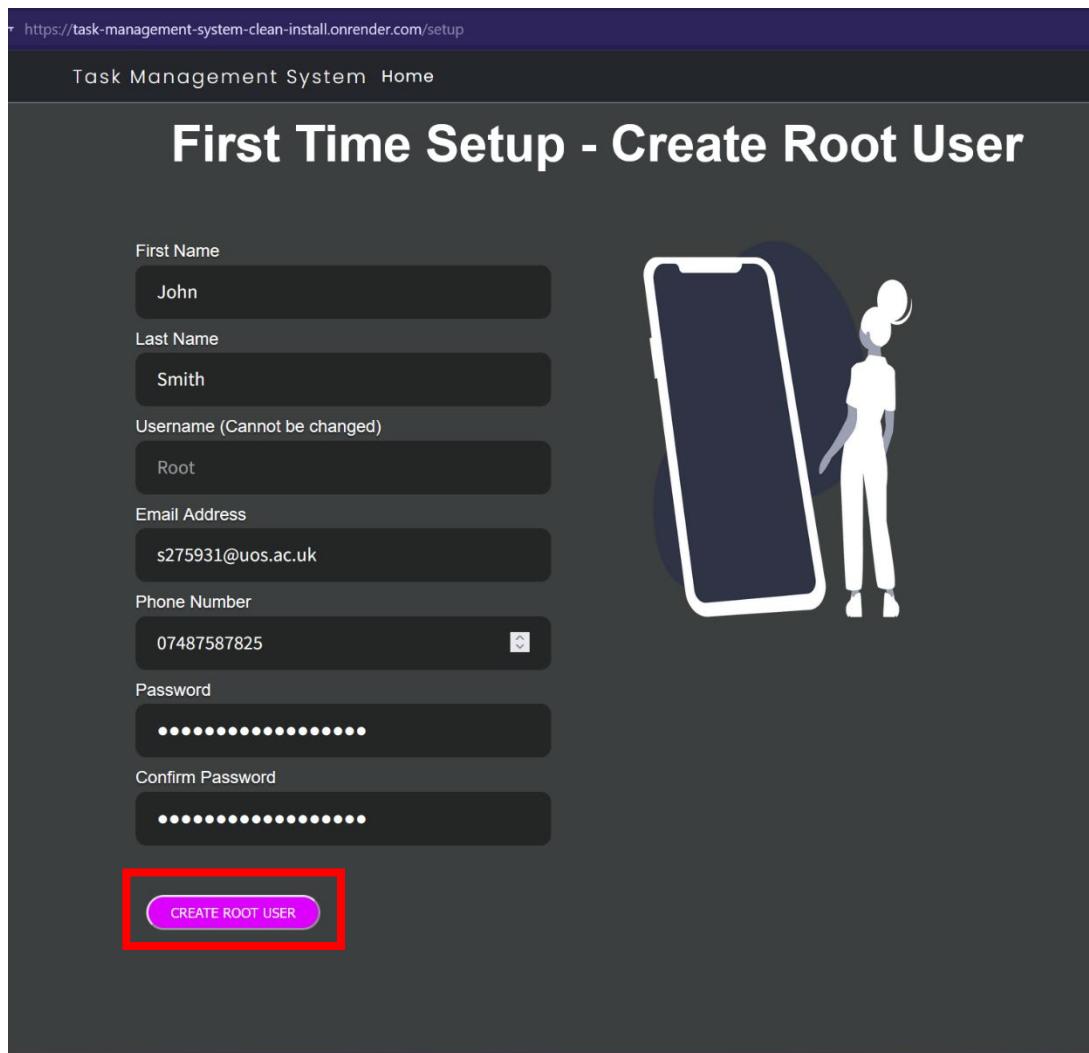


Figure 1 - Task Management System Setup Page

Fill in the fields in the form then click the *Create Root User* button. The Root username cannot be changed. The password must be a minimum length of 10 characters, have at least one capital letter, one lowercase letter, one number and one special character (ASCII characters only). The password must not be a common password. See the screenshot below for an example.



The screenshot shows a web browser window with the URL <https://task-management-system-clean-install.onrender.com/setup>. The page title is "Task Management System Home". The main heading is "First Time Setup - Create Root User". On the left, there is a form with the following fields:

- First Name: John
- Last Name: Smith
- Username (Cannot be changed): Root
- Email Address: s275931@uos.ac.uk
- Phone Number: 07487587825
- Password: (redacted)
- Confirm Password: (redacted)

A large graphic of a smartphone is displayed on the right, with a white silhouette of a person standing next to it. At the bottom of the form is a pink button labeled "CREATE ROOT USER", which is highlighted with a red rectangular border.

Figure 2 - Task Management System Setup Form Example

The Root user will then be redirected to the Home page and will be logged in to the Task Management System. The root user will automatically receive an email with their password sent to the email address entered in the setup form.

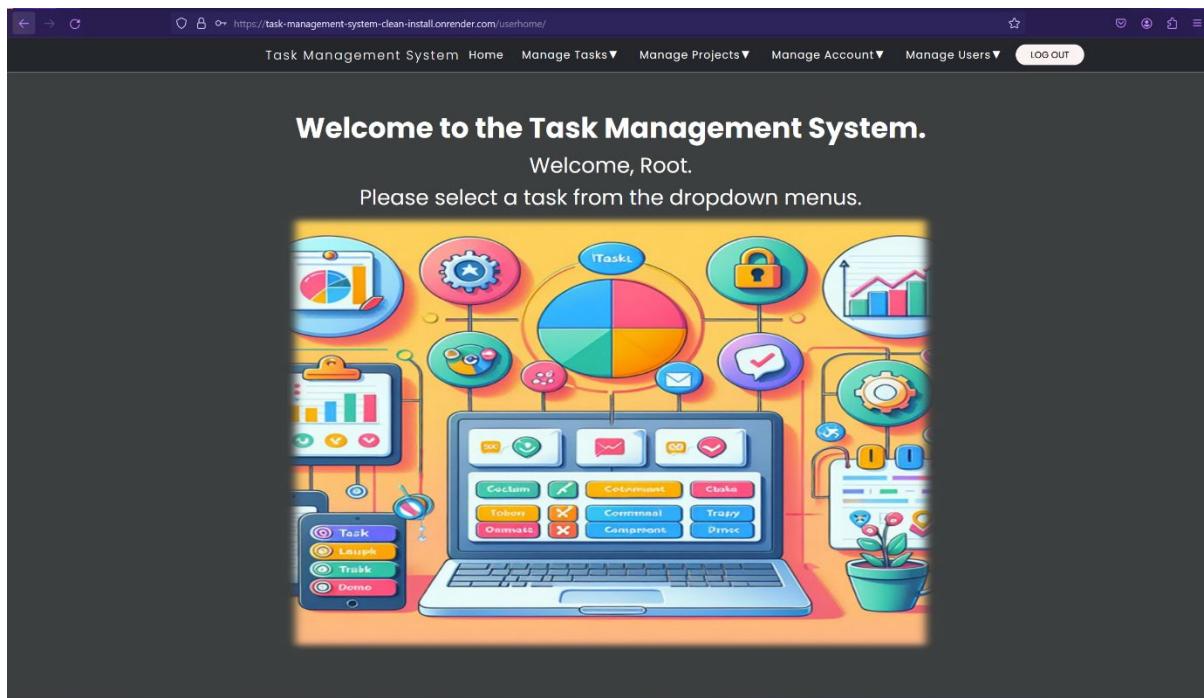


Figure 3 - Task Management System Home Page when logged in.

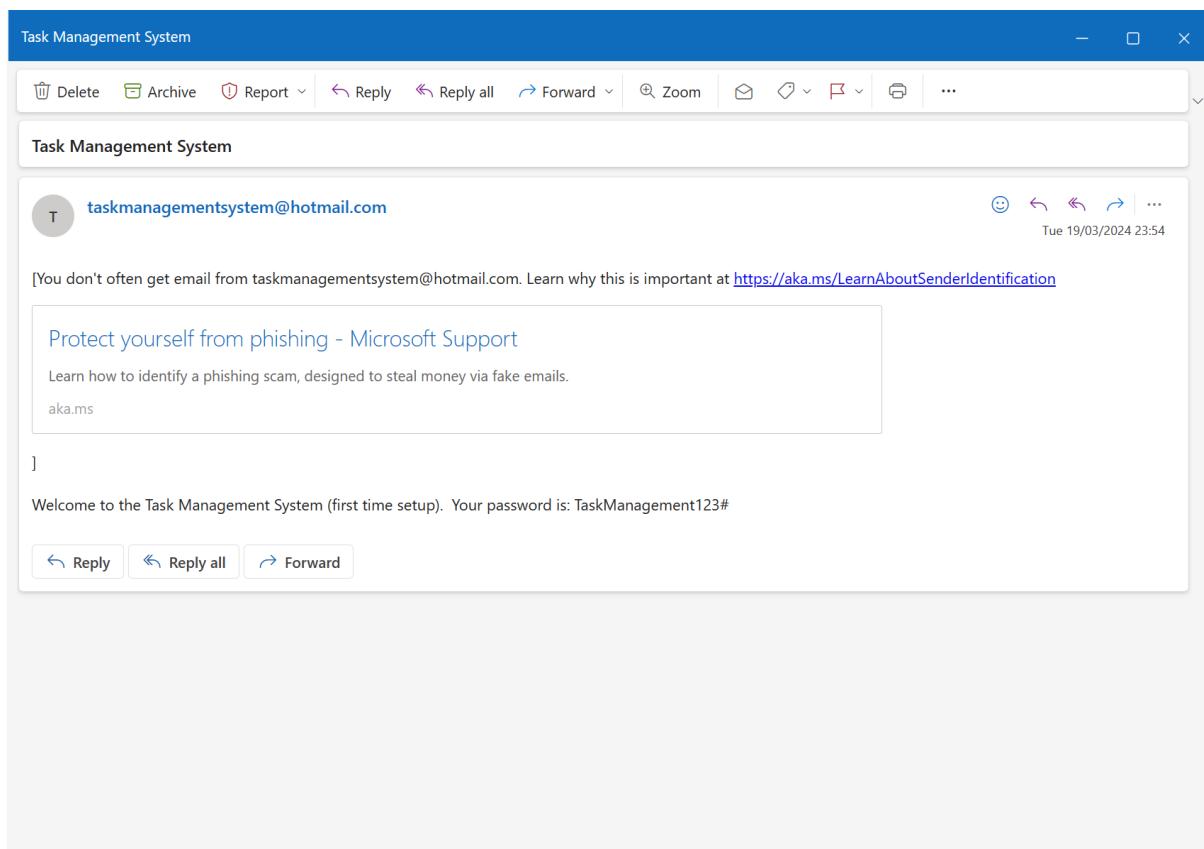


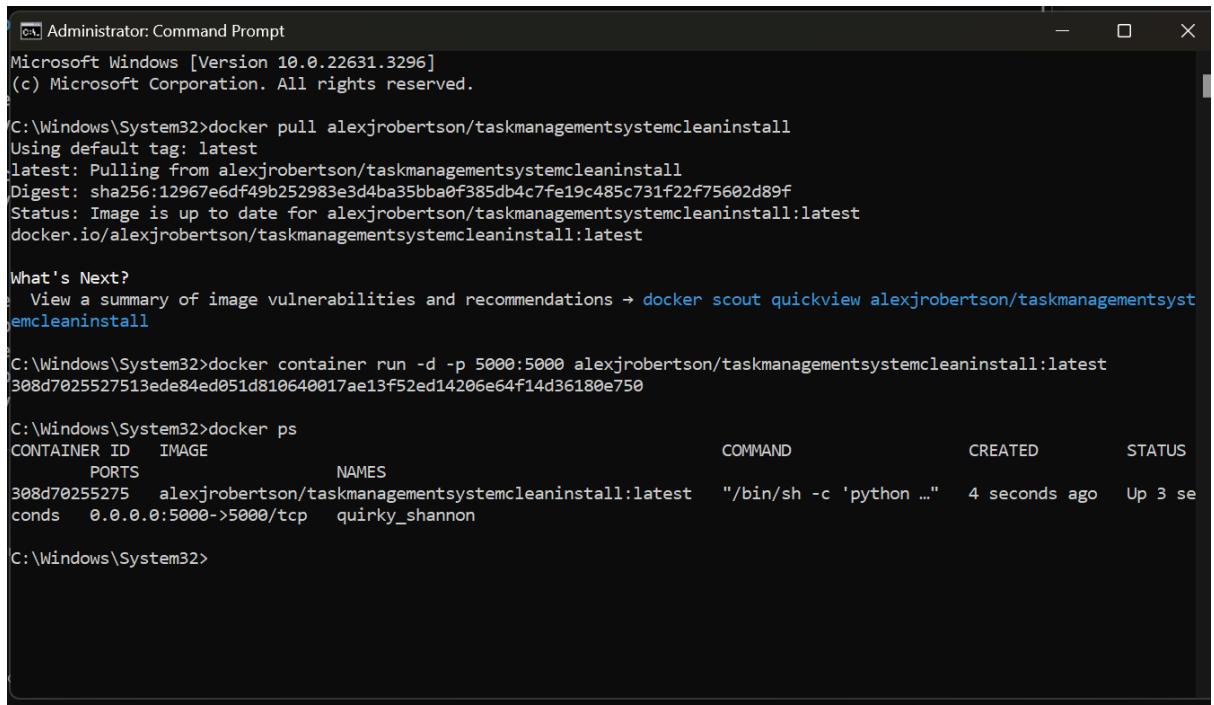
Figure 4 - Automatic email containing the Root User's password.

The Task Management System is now ready for use.

Installation and Setup using Docker Deployment

These are instructions on how to install and set up the Task Management System as a Docker deployment.

1. Docker MUST be installed prior to installation and setup of the Task Management System and your computer must meet the system requirements to run Docker. For instructions, refer to the Docker documentation (<https://docs.docker.com/manuals/>) and download Docker here: <https://www.docker.com/products/docker-desktop/> .
2. Go to Docker hub to copy the pull command. For clean installation: <https://hub.docker.com/r/alexjrobertson/taskmanagementsystemcleaninstall> , for Task Management System with example data, users, tasks and projects: <https://hub.docker.com/r/alexjrobertson/taskmanagementsystem> .
3. Copy the following Docker pull command: `docker pull alexjrobertson/taskmanagementsystemcleaninstall` into the computer terminal, Command Prompt or PowerShell for download for a clean installation of the Task Management System container. Copy the following Docker pull command: `docker pull alexjrobertson/taskmanagementsystem` into the computer terminal, Command Prompt or PowerShell for the Task Management System Container with example data, users, tasks, and projects. When using Docker on Linux, you may be required to type ‘sudo’ before entering the Docker commands or else you may get a ‘permission denied’ or ‘access denied’ error. You may also be prompted to enter your computer password (must be an administrator, root or superuser password) into the Linux terminal. This applies to all Docker commands when using Linux.
4. Enter the following Docker command to run the container: `docker container run -d -p 5000:5000 alexjrobertson/taskmanagementsystemcleaninstall:Latest` for the clean installation and `docker container run -d -p 5000:5000 alexjrobertson/taskmanagementsystem:Latest` for the Task Management System with example data, users, tasks, and projects. The port number in red can be changed to suit the specific need of the host running the Task Management System. This is especially important if you have multiple containers running or multiple applications using localhost as two applications or containers cannot share the same port.



```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>docker pull alexjrobertson/taskmanagementsystemcleaninstall
Using default tag: latest
latest: Pulling from alexjrobertson/taskmanagementsystemcleaninstall
Digest: sha256:12967e6df49b252983e3d4ba35bba0f385db4c7fe19c485c731f22f75602d89f
Status: Image is up to date for alexjrobertson/taskmanagementsystemcleaninstall:latest
docker.io/alexjrobertson/taskmanagementsystemcleaninstall:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview alexjrobertson/taskmanagementsystemcleaninstall

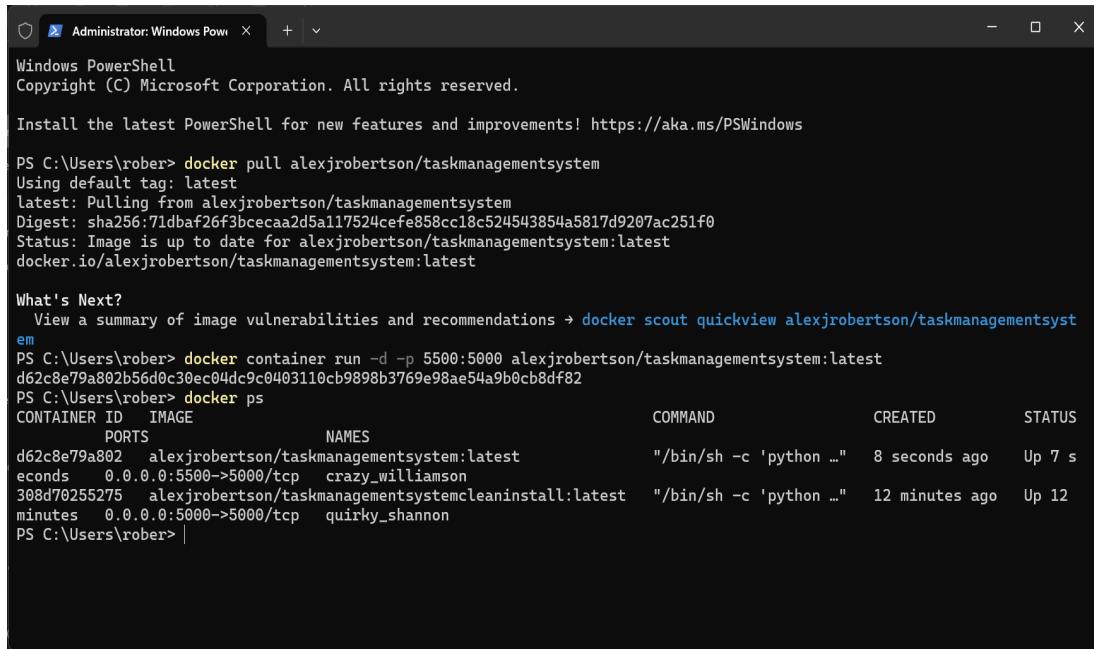
C:\Windows\System32>docker container run -d -p 5000:5000 alexjrobertson/taskmanagementsystemcleaninstall:latest
308d7025527513ede84ed051d810640017ae13f52ed14206e64f14d36180e750

C:\Windows\System32>docker ps
CONTAINER ID   IMAGE               COMMAND                  CREATED        STATUS
              PORTS NAMES
308d70255275   alexjrobertson/taskmanagementsystemcleaninstall:latest   "/bin/sh -c 'python ...'"   4 seconds ago   Up 3 seconds
  0.0.0.0:5000->5000/tcp   quirky_shannon

C:\Windows\System32>

```

Figure 5 - Example of downloading and running the Task Management System container using Docker commands in the Command Prompt.



```

Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rober> docker pull alexjrobertson/taskmanagementsystem
Using default tag: latest
latest: Pulling from alexjrobertson/taskmanagementsystem
Digest: sha256:71dbaf26f3bcecaa2d5a117524cefef858cc18c524543854a5817d9207ac251f0
Status: Image is up to date for alexjrobertson/taskmanagementsystem:latest
docker.io/alexjrobertson/taskmanagementsystem:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview alexjrobertson/taskmanagementsystem

PS C:\Users\rober> docker container run -d -p 5500:5000 alexjrobertson/taskmanagementsystem:latest
d62c8e79a802b56d0c30ec04dc9c0403110cb9898b3769e98ae54a9b0cb8df82
PS C:\Users\rober> docker ps
CONTAINER ID   IMAGE               COMMAND                  CREATED        STATUS
              PORTS NAMES
d62c8e79a802   alexjrobertson/taskmanagementsystem:latest   "/bin/sh -c 'python ...'"   8 seconds ago   Up 7 seconds
  0.0.0.0:5500->5000/tcp   crazy_williamson
308d70255275   alexjrobertson/taskmanagementsystemcleaninstall:latest   "/bin/sh -c 'python ...'"   12 minutes ago  Up 12 minutes
  0.0.0.0:5000->5000/tcp   quirky_shannon

PS C:\Users\rober>

```

Figure 6 - Example of downloading the Task Management System image and running the Task Management System container using Docker commands in the terminal (Windows PowerShell)

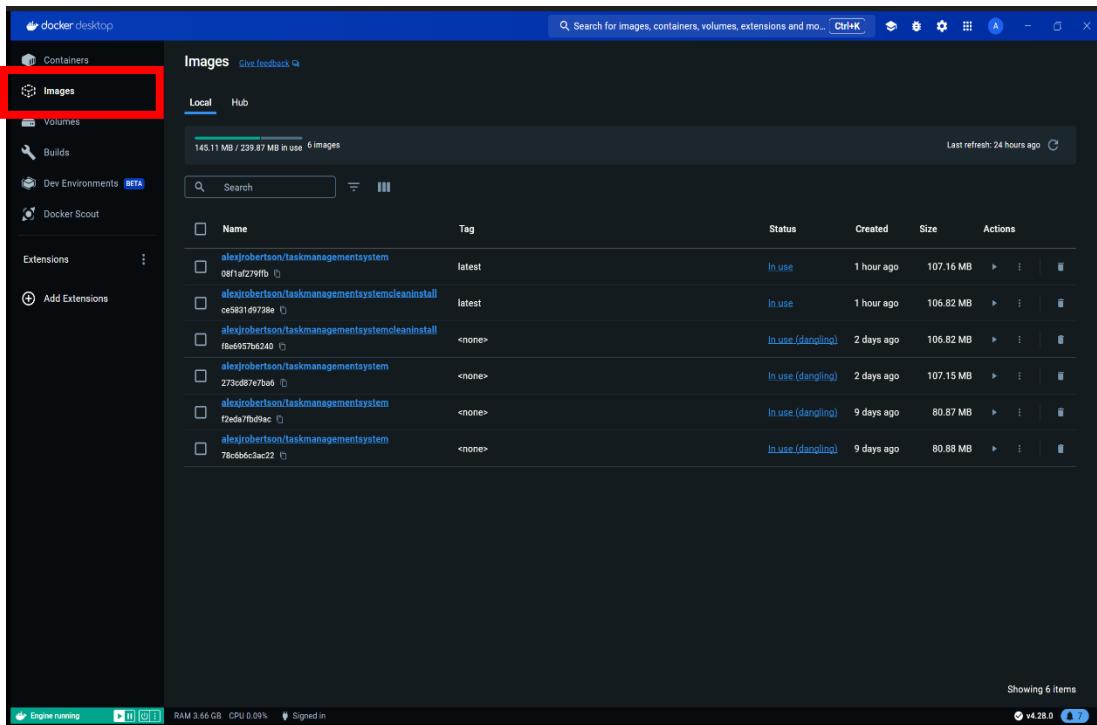


Figure 7 - Task Management System images visible on Docker Desktop.

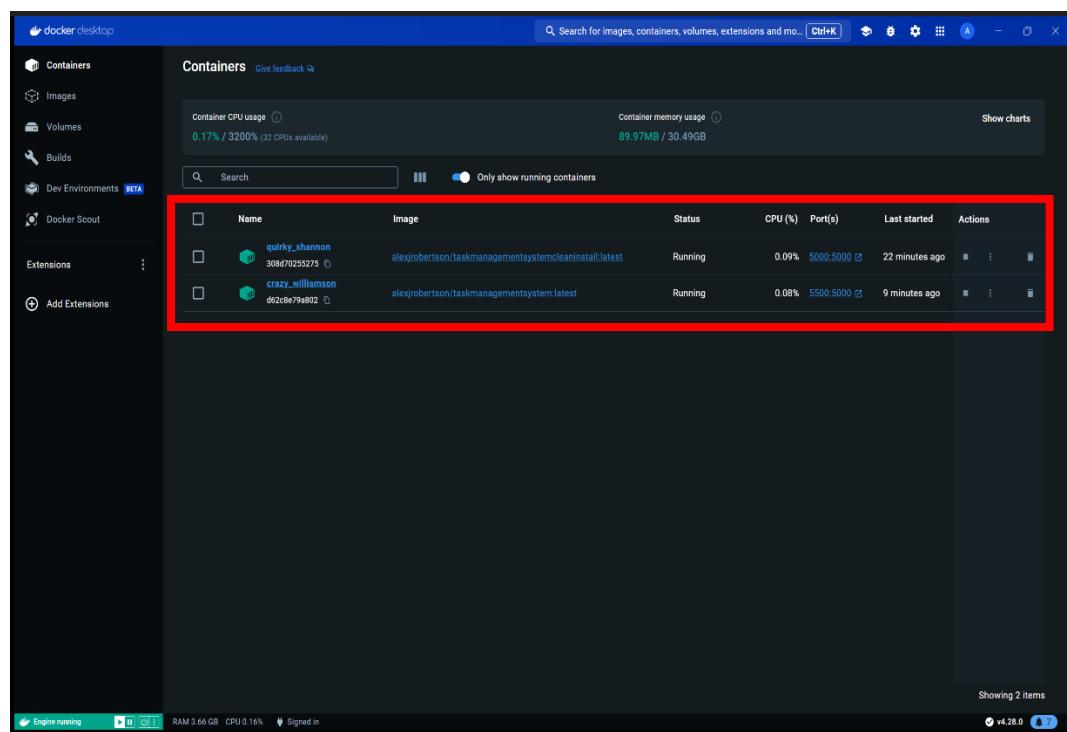


Figure 8 - Running Task Management System containers visible on Docker Desktop.

5. You can also run the Task Management System containers by clicking the play ► buttons beneath the Actions page below the Actions tab on the Images page. Then open Optional Settings in the popup menu that appears and assign a port to the container and then click Run. See the screenshot below for an example.

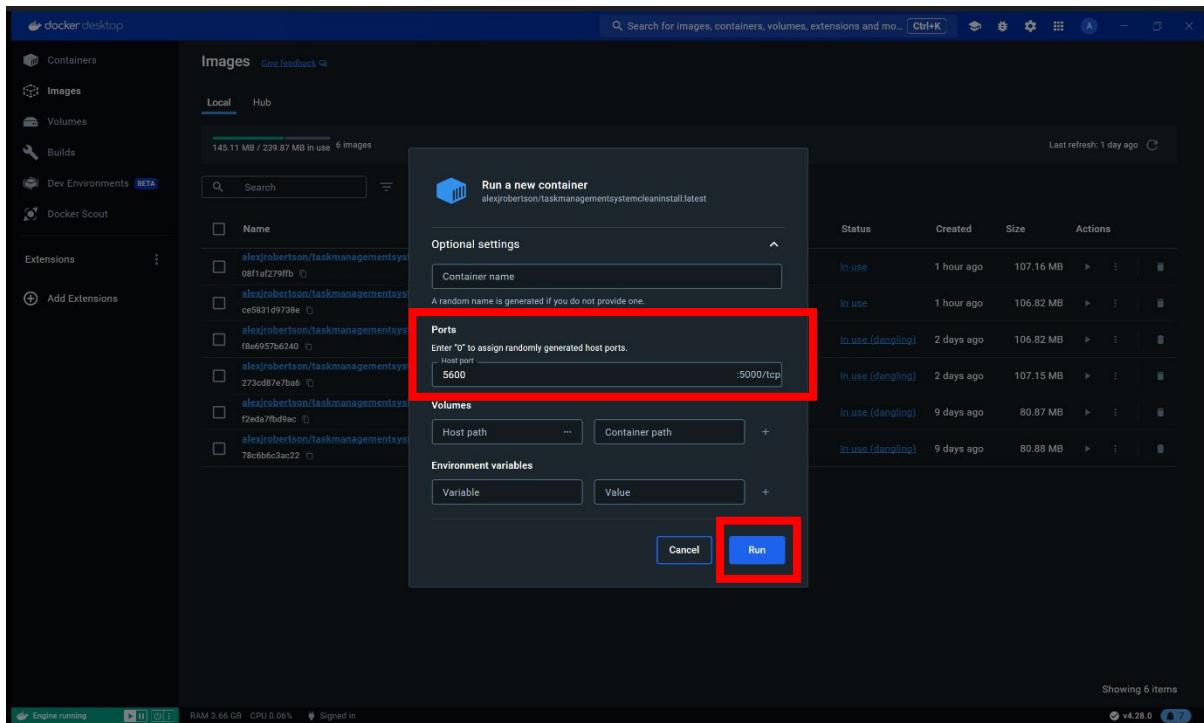


Figure 9 - Running the Tasks Management System container from Docker Desktop GUI. Make sure to assign a port number.

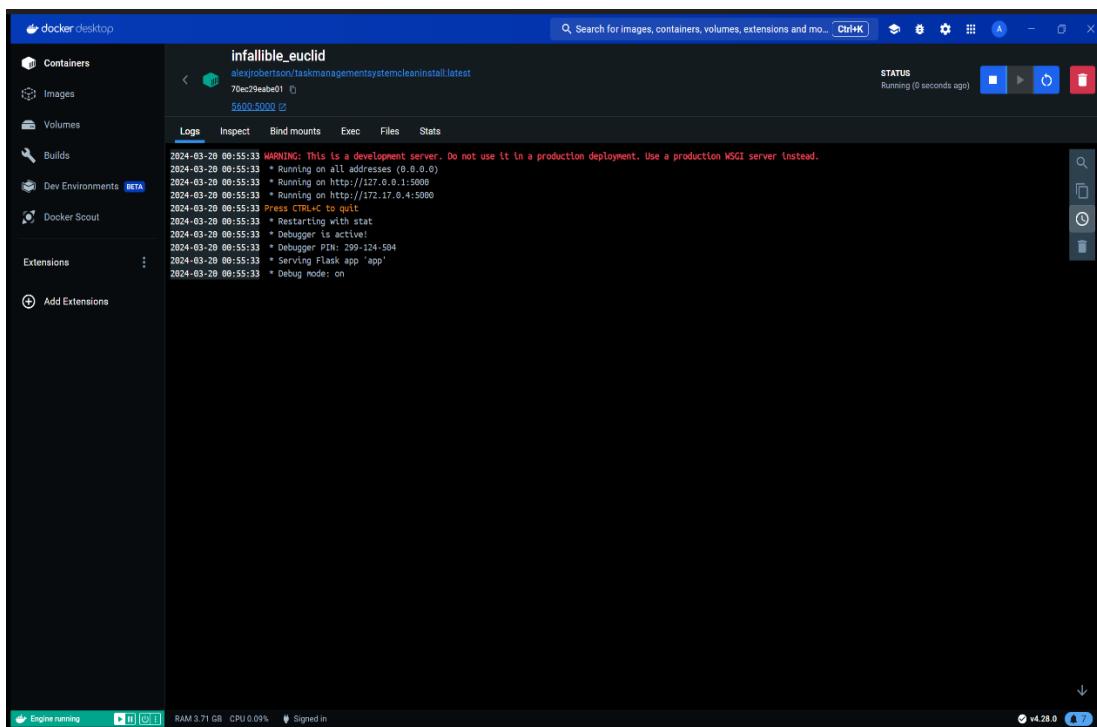


Figure 10 - Docker Desktop after running the Task Management System container.

6. Type `localhost:<port number>` e.g. `localhost:5000` or `localhost:5500` into the web browser address bar and press enter. The port you enter must be the same as the one you entered in the `docker container run` command. The Task Management System Setup page or Landing page (Home / index.html) will appear.

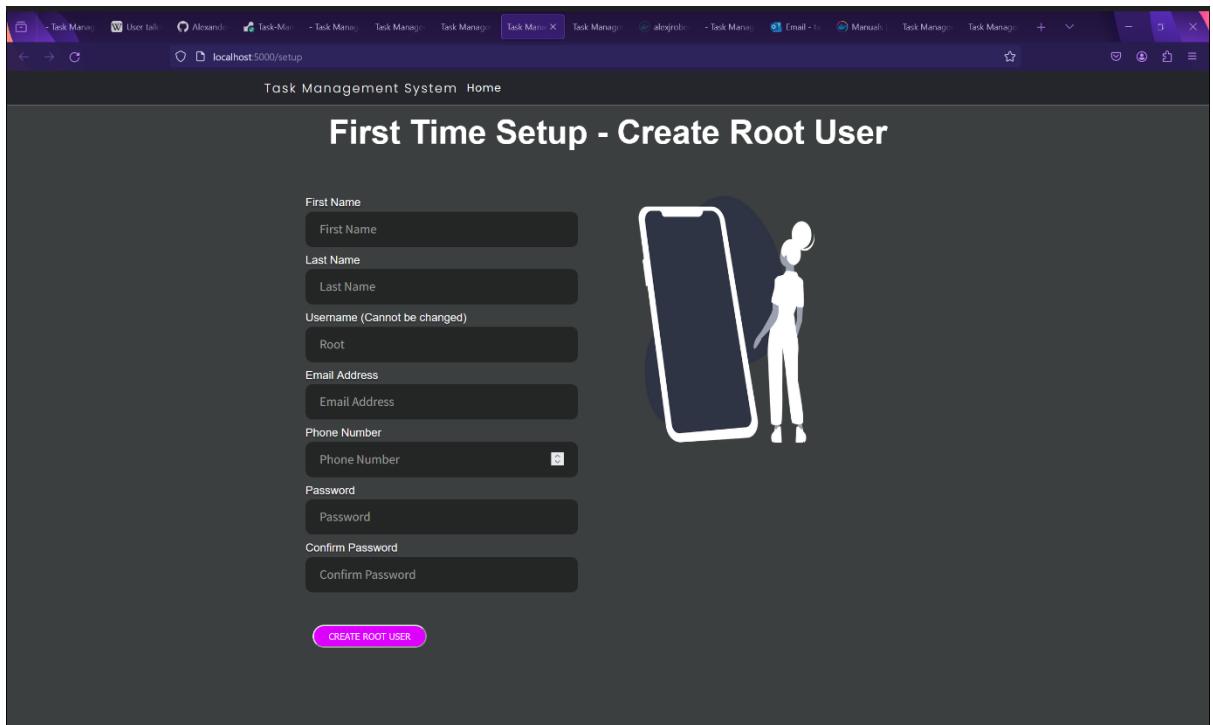
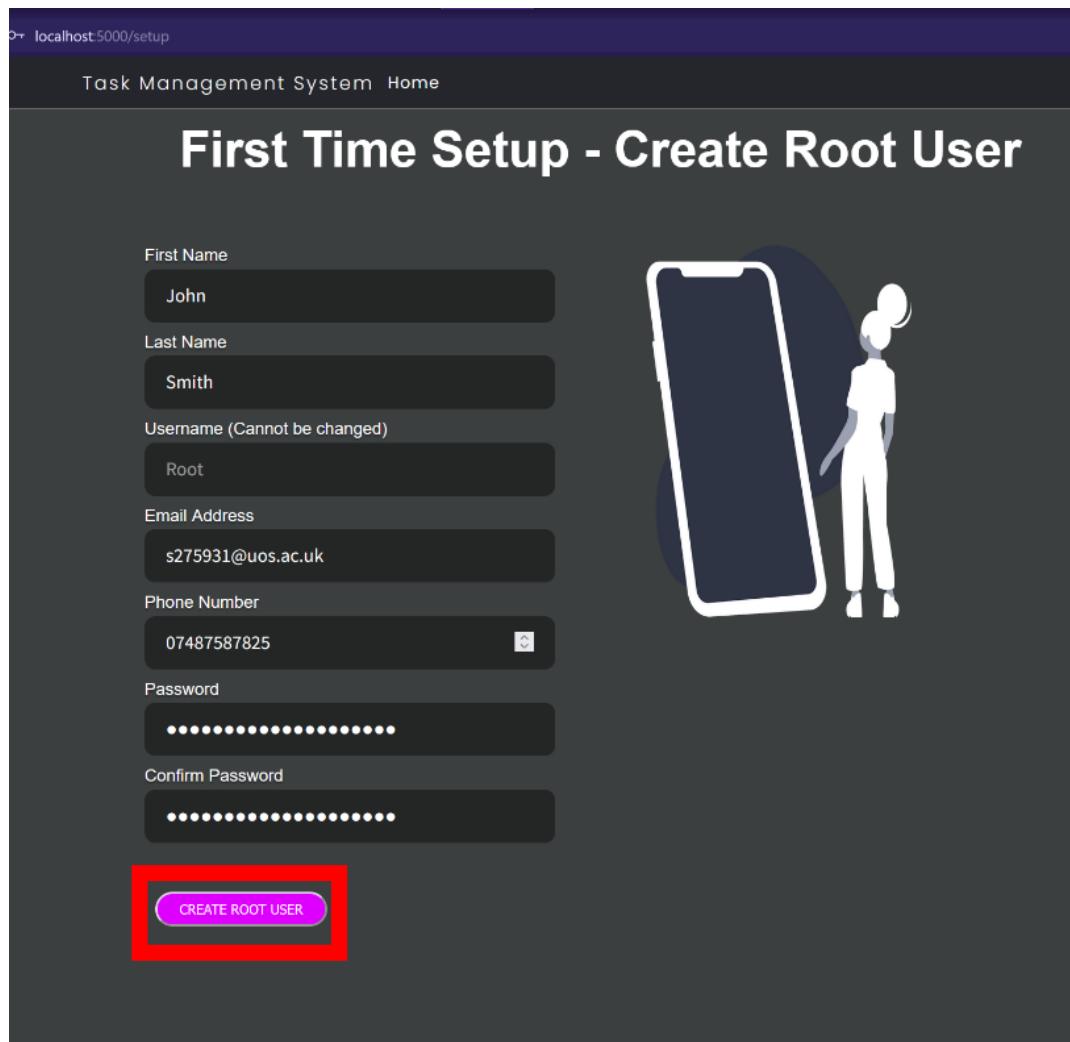


Figure 11 - The Task Management System Setup page is loaded when you enter localhost:<container port number> e.g. localhost:5000 into the address bar in your web browser.

7. Fill in the fields in the form then click the *Create Root User* button. The Root username cannot be changed. The password must be a minimum length of 10 characters, have at least one capital letter, one lowercase letter, one number and one special character (ASCII characters only). The password must not be a common password. See the screenshot below for an example.



8. You will be redirected to the homepage and be logged in.



Figure 12 - The Root User is redirected to the home page and logged in after setup.

9. You will receive an automatic email with your password.

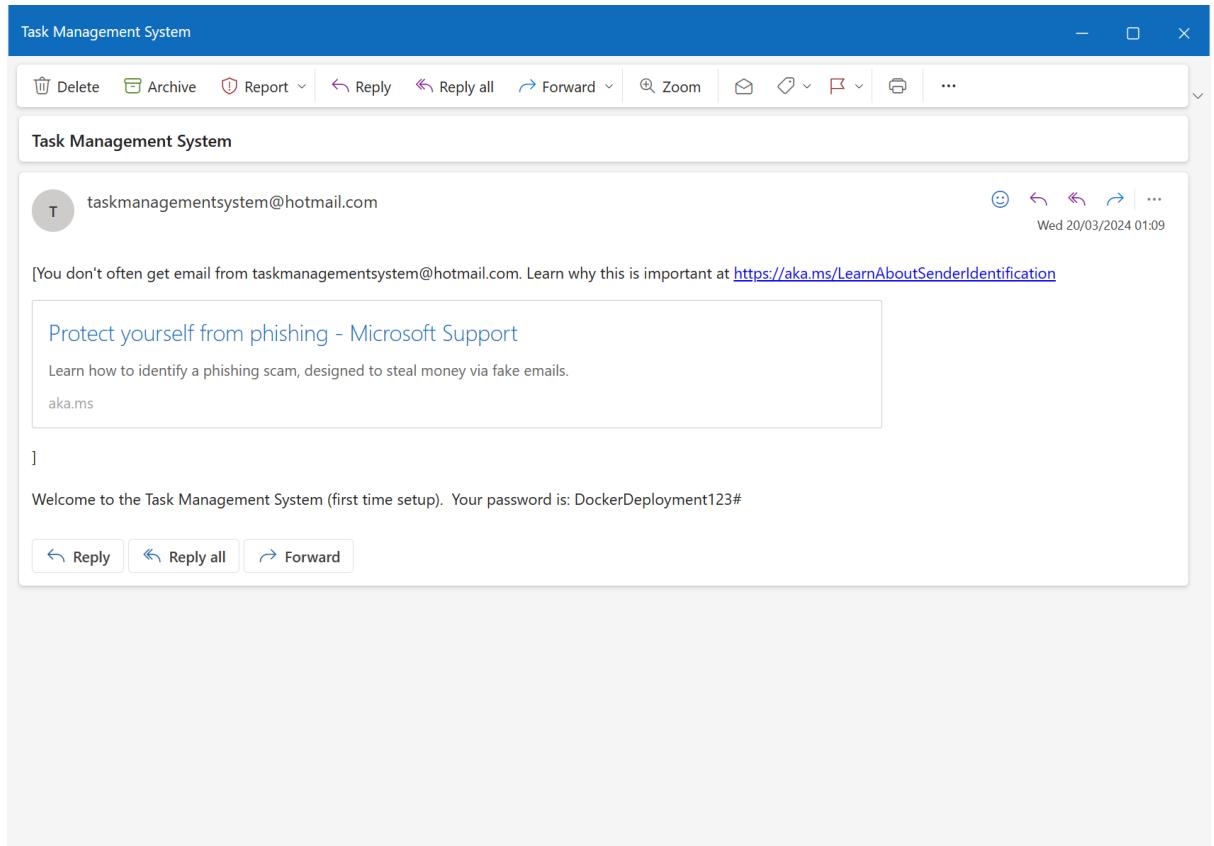


Figure 13 - Email containing Root User's password.

10. The Task Management System is ready to use.

11. For the Task Management System with example data, users, tasks, and projects:

- a. The username for the Root user is Root and the password is TaskManagement123#
- b. The username for an administrator is barneyPurpleDinosaur and the password is BabyBop123#
- c. The username for a standard user is Birmingham888 and the password is SouthSide123@

12. To stop the Task Management System Docker containers, Click the Stop ■ buttons below the Actions tab on the Containers page on Docker Desktop or type docker `stop <container_id>` where `<container_id>` is the ID number or name of the container. See the two screenshots below for an example.

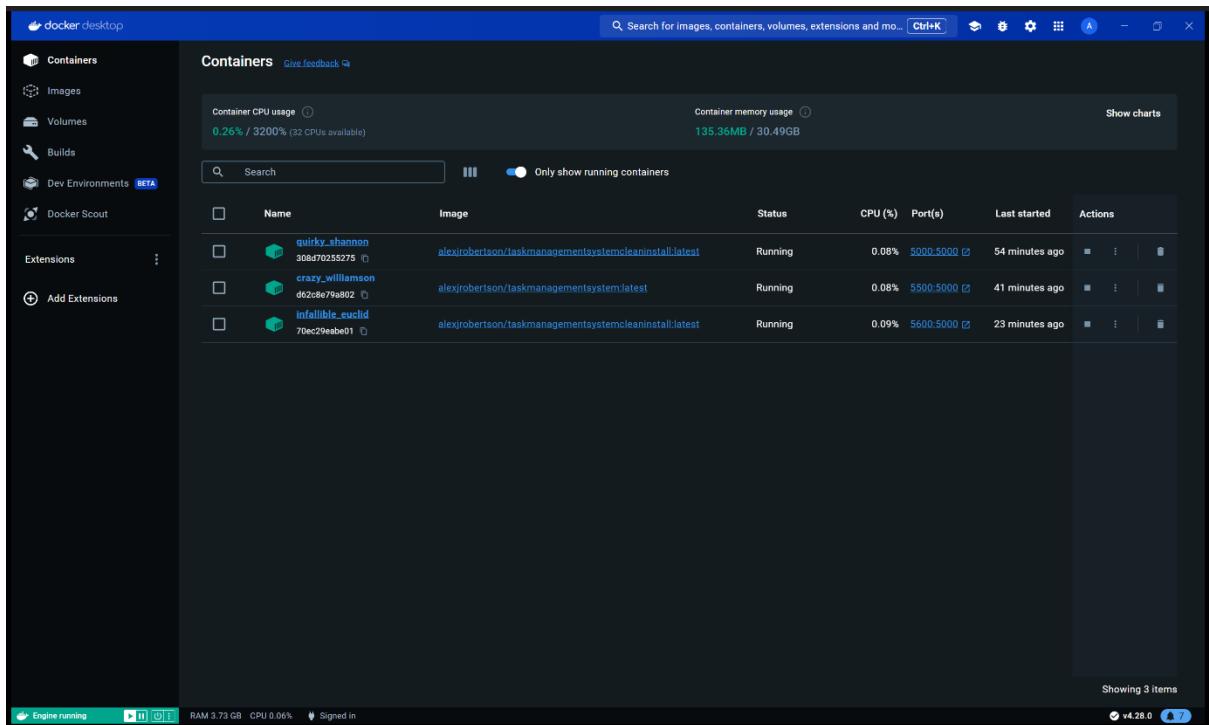


Figure 14 - Stopping Task Management System containers in Docker Desktop.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
ca6ab2c711b9 alexrobertson/taskmanagementsystem:latest "/bin/sh -c 'python ...'" 24 seconds ago Up 23 seconds 0.0.0.0:5500->5000/tcp exciting_rosalind
99a4e5aa0153 alexrobertson/taskmanagementsystemcleaninstall:latest "/bin/sh -c 'python ...'" 55 seconds ago Up 53 seconds 0.0.0.0:5000->5000/tcp zen_goldwasser

C:\Windows\System32>docker stop ca6ab2c711b9
ca6ab2c711b9

C:\Windows\System32>docker stop zen_goldwasser
zen_goldwasser

C:\Windows\System32>docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
C:\Windows\System32>
```

Figure 15 - Using docker stop commands using the Command Prompt / PowerShell / Terminal

Installation and Setup Using Source Code

Instructions to download the Task Management System source code from GitHub repository and build and deploy the application from scratch. Using this method, you can modify the source code to suit your organisation's needs such as modifying, adding, or removing features, changing the architecture, password requirements, database design, automatic email messages and domain, user interface design or experiment with new features. You can also choose a deployment platform of your own choice however this may require you to make modifications to the source code and files.

WARNING: ONLY MODIFY THE SOURCE CODE IF YOU HAVE EXPERIENCE WITH AND UNDERSTAND PYTHON 3, SQL, SQLITE, HTML, CSS, JAVASCRIPT, FLASK, DOCKER AND DOCKERFILES, GUNICORN AND GIT. EDITING THESE FILES WITHOUT THE RIGHT KNOWHOW MAY DAMAGE THE TASK MANAGEMENT SYSTEM, RENDERING THE APPLICATION INOPERABLE AND/OR CORRUPT THE DATABASE RESULTING IN PERMANENT DATA LOSS! ALWAYS BACK UP THE ORIGINAL APPLICATION AND USE VERSION CONTROL ON YOUR OWN GIT REPOSITORY IN CASE THE APPLICATION GETS DAMAGED. THE SOFTWARE AUTHOR CANNOT BE HELD RESPONSIBLE FOR DAMAGE CAUSED BY A THIRD PARTY MODIFYING THE SOURCE CODE.

Prerequisites

To modify, build and deploy the Task Management System from scratch, you will need the following software installed on your computer:

- Windows 10, Windows 11, macOS 12.0 (Monterey) or a supported distribution of Linux (Ubuntu, Debian, RHEL etc.)
- Windows Subsystem for Linux (WSL) or Hyper-V features enabled if using Windows and a compatible version of Linux installed in WSL (see above)
- PyCharm IDE
- Python 3.12 interpreter
- SQLite Studio
- Git
- Docker
- Compatible web browser (Firefox, Google Chrome, Microsoft Edge, Safari etc.)

Downloading from the GitHub repository

Firstly, you will need to download the Task Management System from the GitHub repository by cloning the repository using the git clone command in the Terminal / PowerShell / Command Prompt.

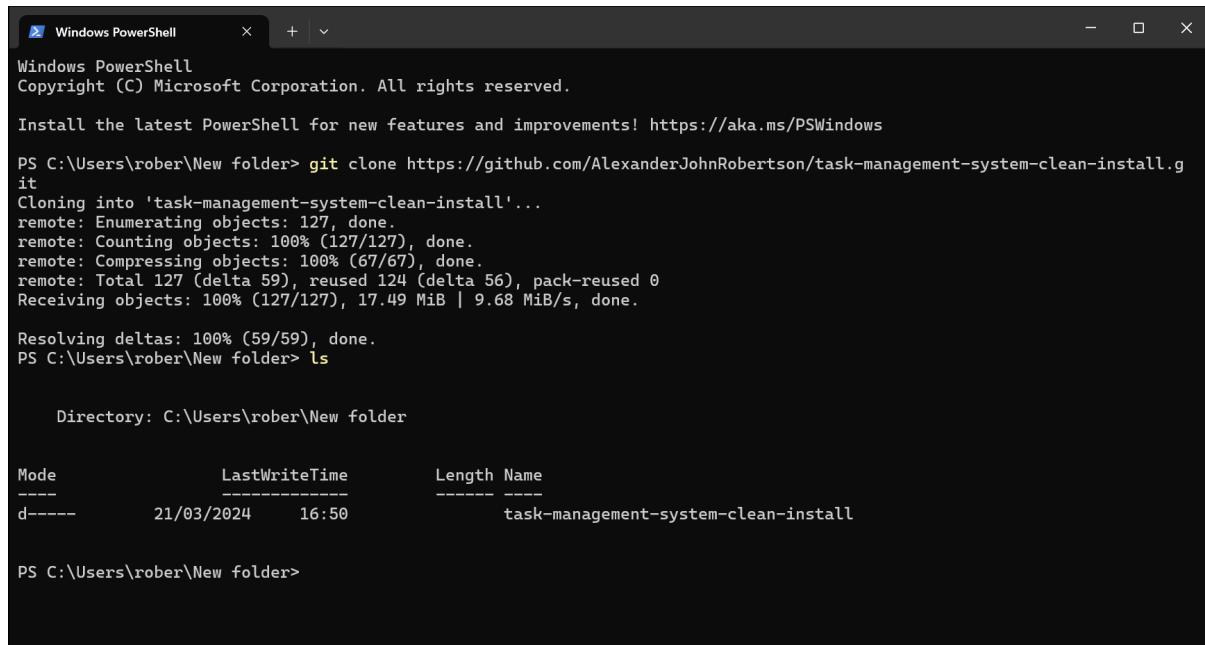
Link to GitHub repository for clean install: <https://github.com/AlexanderJohnRobertson/task-management-system-clean-install.git>

Link to GitHub repository for Task Management System with example data, users, tasks and projects: <https://github.com/AlexanderJohnRobertson/task-management-system.git> (The username for the Root user is Root and the password is TaskManagement123# , The username for an administrator is barneyPurpleDinosaur and the password is BabyBop123# , The username for a standard user is Birmingham888 and the password is SouthSide123@).

Enter the following git command into the computer Terminal / PowerShell / Command Prompt to clone and download the Task Management System as a clean install: `git clone https://github.com/AlexanderJohnRobertson/task-management-system-clean-install.git`

Enter the following git command into the computer Terminal / PowerShell / Command Prompt to clone and download the Task Management System with example data, users, tasks and projects: `git clone https://github.com/AlexanderJohnRobertson/task-management-system.git`

S275931



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rober\New folder> git clone https://github.com/AlexanderJohnRobertson/task-management-system-clean-install.git
Cloning into 'task-management-system-clean-install'...
remote: Enumerating objects: 127, done.
remote: Counting objects: 100% (127/127), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 127 (delta 59), reused 124 (delta 56), pack-reused 0
Receiving objects: 100% (127/127), 17.49 MiB | 9.68 MiB/s, done.

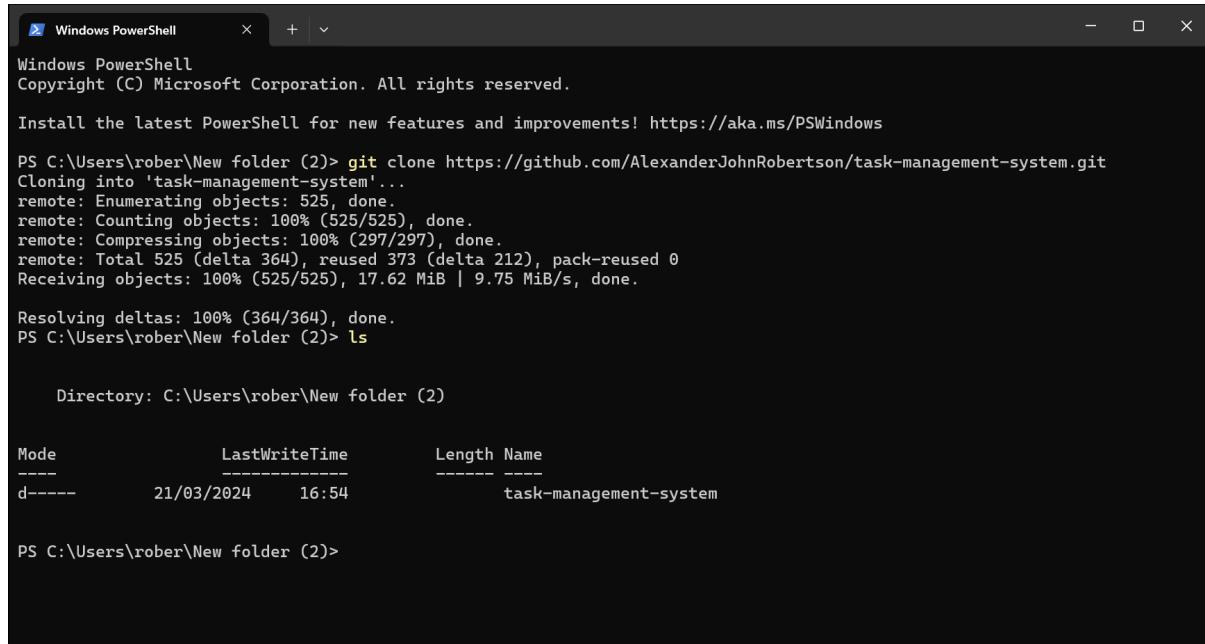
Resolving deltas: 100% (59/59), done.
PS C:\Users\rober\New folder> ls

Directory: C:\Users\rober\New folder

Mode                LastWriteTime         Length Name
----                -----          ---- -  
d-----        21/03/2024      16:50    task-management-system-clean-install

PS C:\Users\rober\New folder>
```

Figure 16 - Screenshot of cloning the Task Management System Clean Install Git repository.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rober\New folder (2)> git clone https://github.com/AlexanderJohnRobertson/task-management-system.git
Cloning into 'task-management-system'...
remote: Enumerating objects: 525, done.
remote: Counting objects: 100% (525/525), done.
remote: Compressing objects: 100% (297/297), done.
remote: Total 525 (delta 364), reused 373 (delta 212), pack-reused 0
Receiving objects: 100% (525/525), 17.62 MiB | 9.75 MiB/s, done.

Resolving deltas: 100% (364/364), done.
PS C:\Users\rober\New folder (2)> ls

Directory: C:\Users\rober\New folder (2)

Mode                LastWriteTime         Length Name
----                -----          ---- -  
d-----        21/03/2024      16:54    task-management-system

PS C:\Users\rober\New folder (2)>
```

Figure 17 - Screenshot of cloning the Task Management System with example data, users, tasks, and projects from the git repository.

Open the PyCharm Project

1. Open PyCharm.
2. On the Welcome to PyCharm screen, select 'Open'.

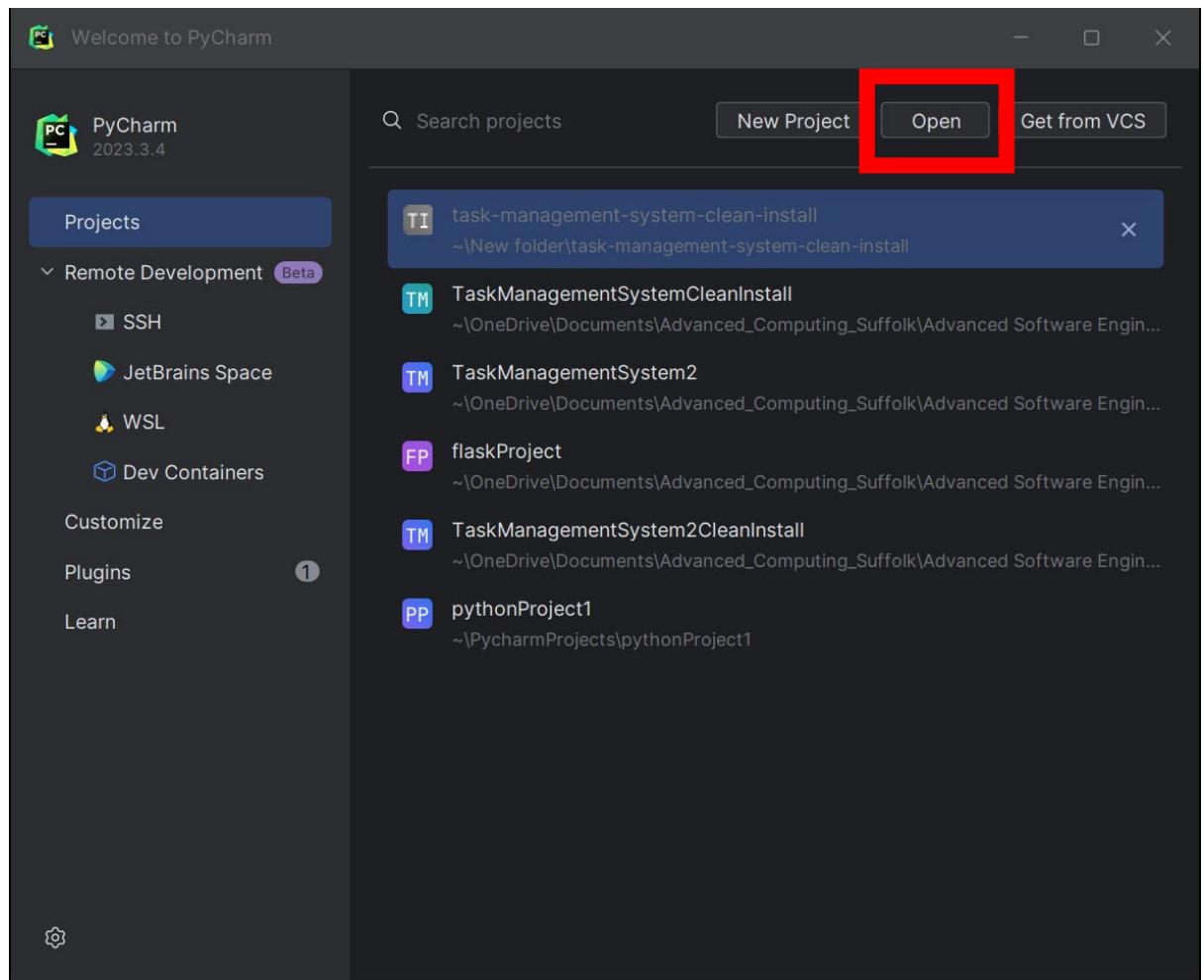


Figure 18 - PyCharm Welcome Screen

3. Navigate to the location of the cloned Git repository and select the “task-management-system-clean-install” or “task-management-system” folder then click “OK”.

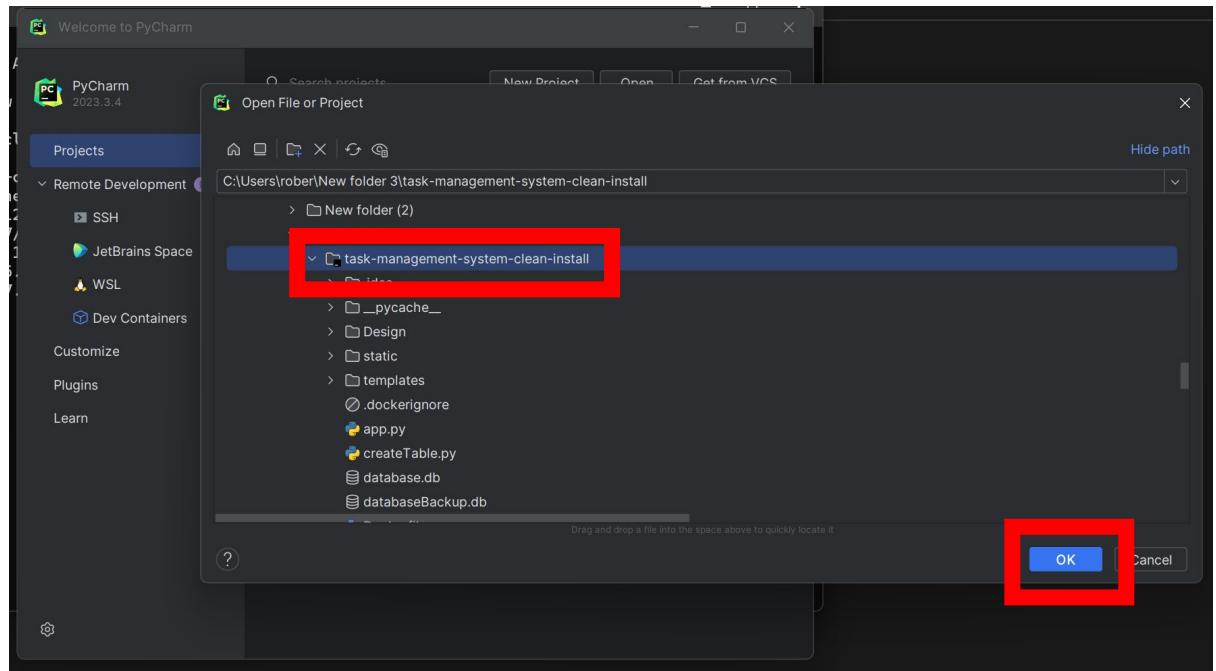


Figure 19 - PyCharm file browser.

4. Click "Trust Project".

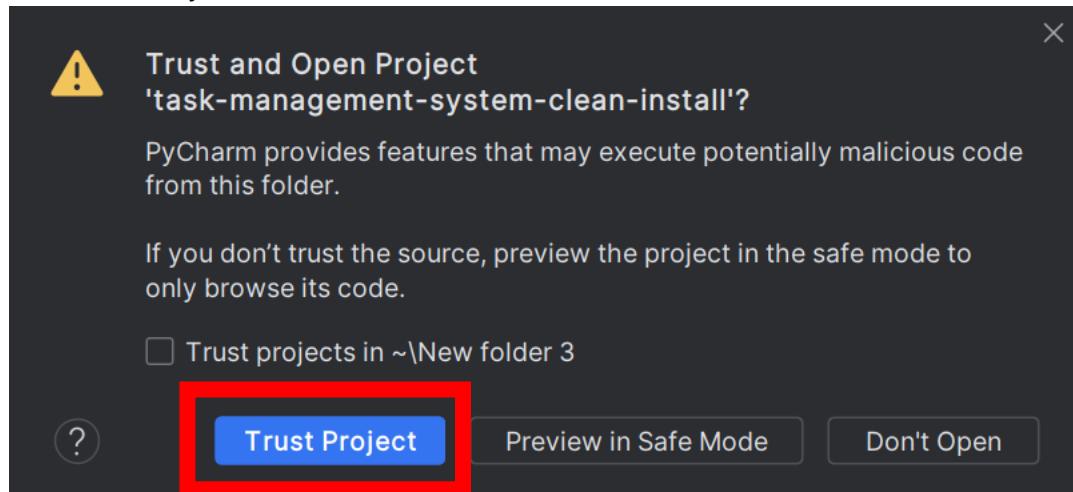


Figure 20 - Select "Trust Project".

5. Wait for PyCharm to load the project and then click on "app.py" – the main Python file in the Task Management System.

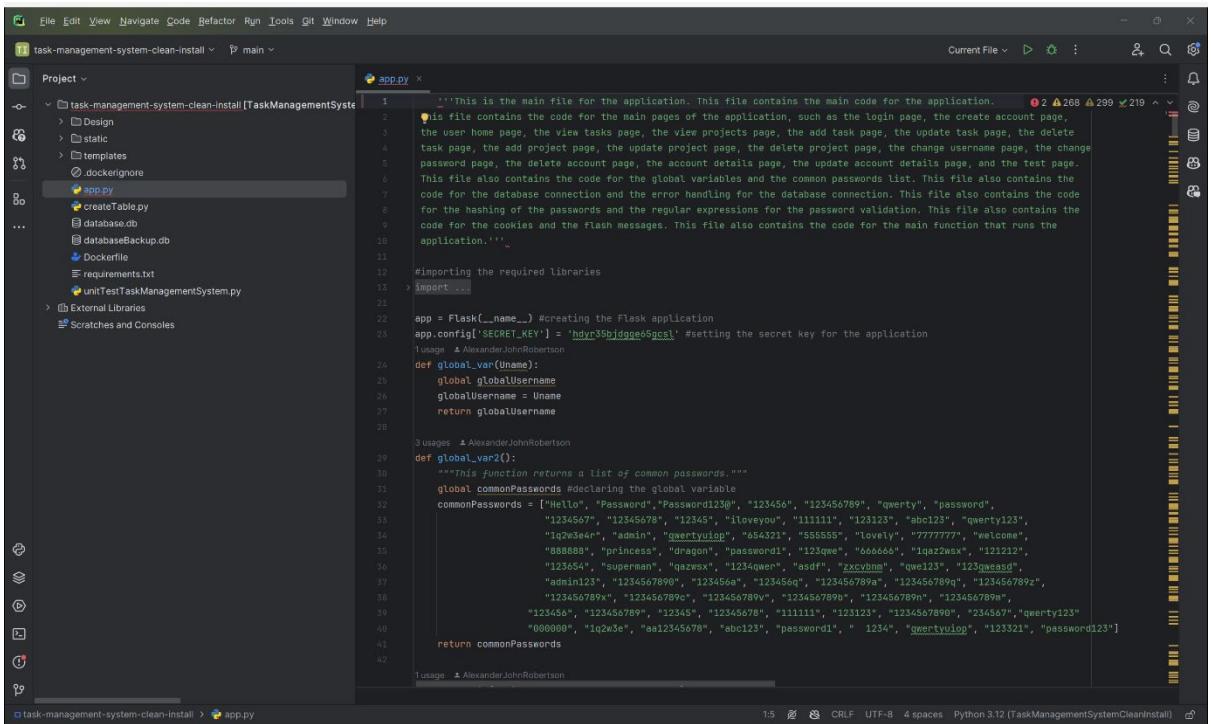


Figure 21 - PyCharm should look like this once it is loaded and app.py is opened.

Setting up the PyCharm environment (Windows)

1. Delete the .venv folder and select “Delete” to confirm.

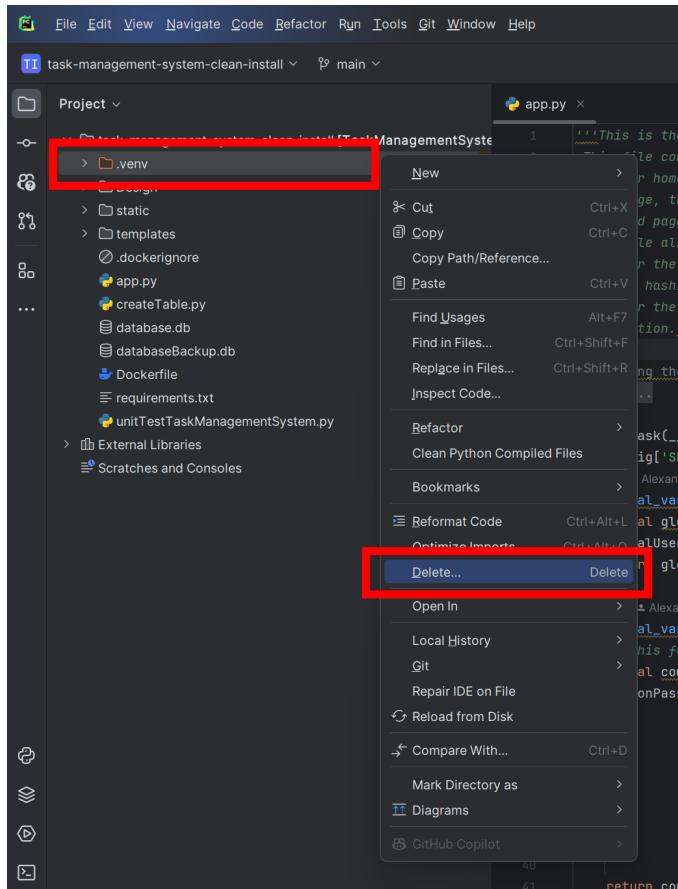


Figure 22 - Deleting the .venv or venv folder

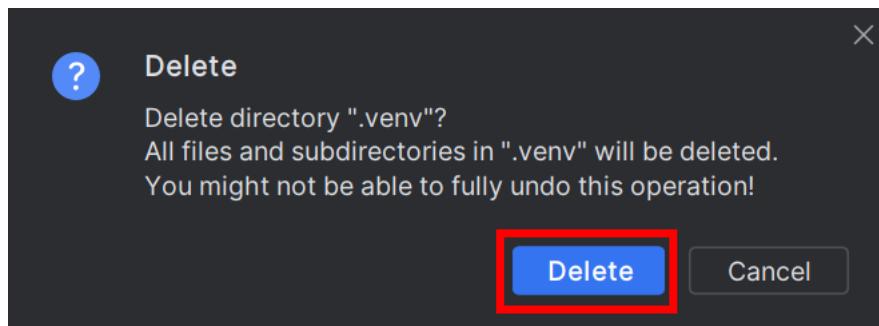


Figure 23 - Click Delete

2. Set up a new Python interpreter for the project.
 - a. Select the Settings (icon in the top right corner of PyCharm IDE) then click "Settings..." in the menu.

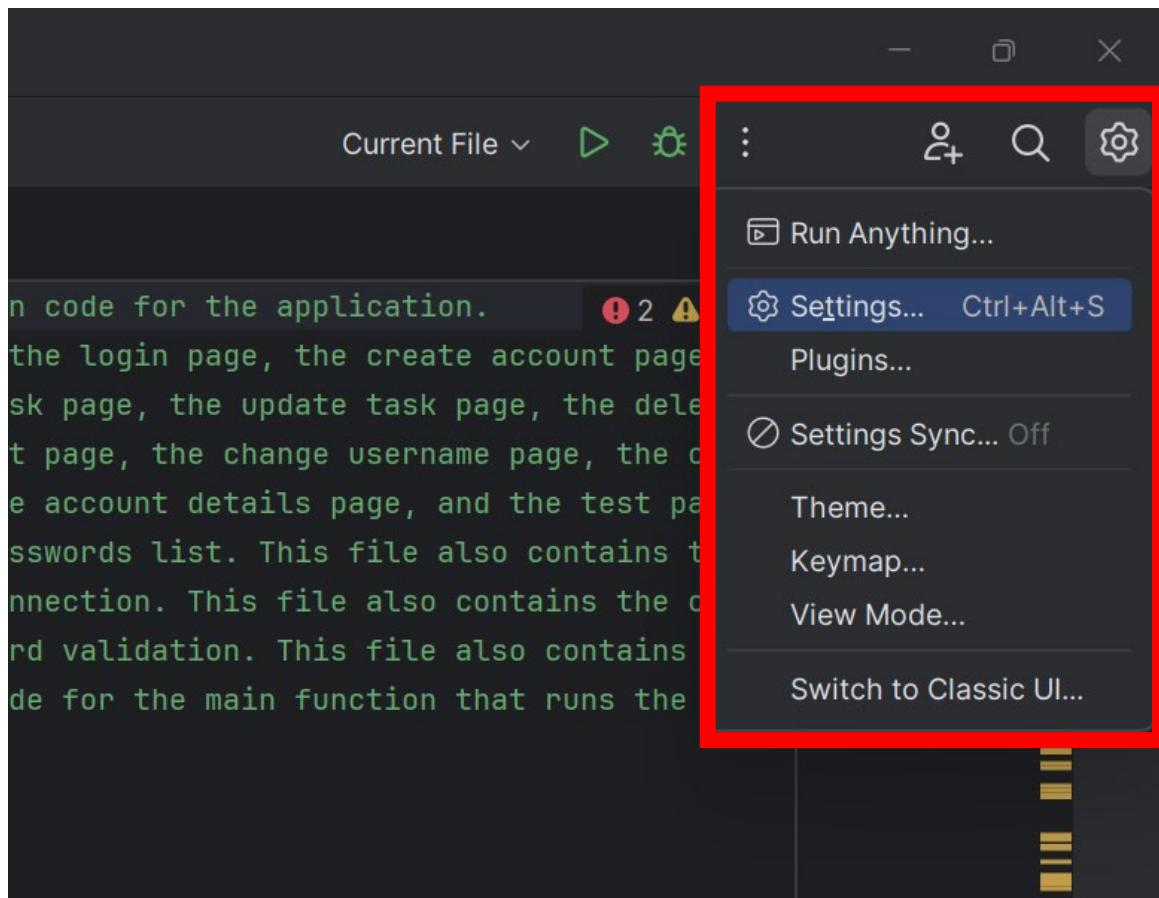


Figure 24 - Selecting the Settings drop down menu.

- b. In the settings menu select “Project: task-management-system” or “Project: task-management-system-clean-install” > “Python Interpreter”.
- c. Select “Add Interpreter” next to “Python Interpreter” and then “Add Local Interpreter...” on the drop-down menu.
- d. Select “No Interpreter” from the drop-down menu.

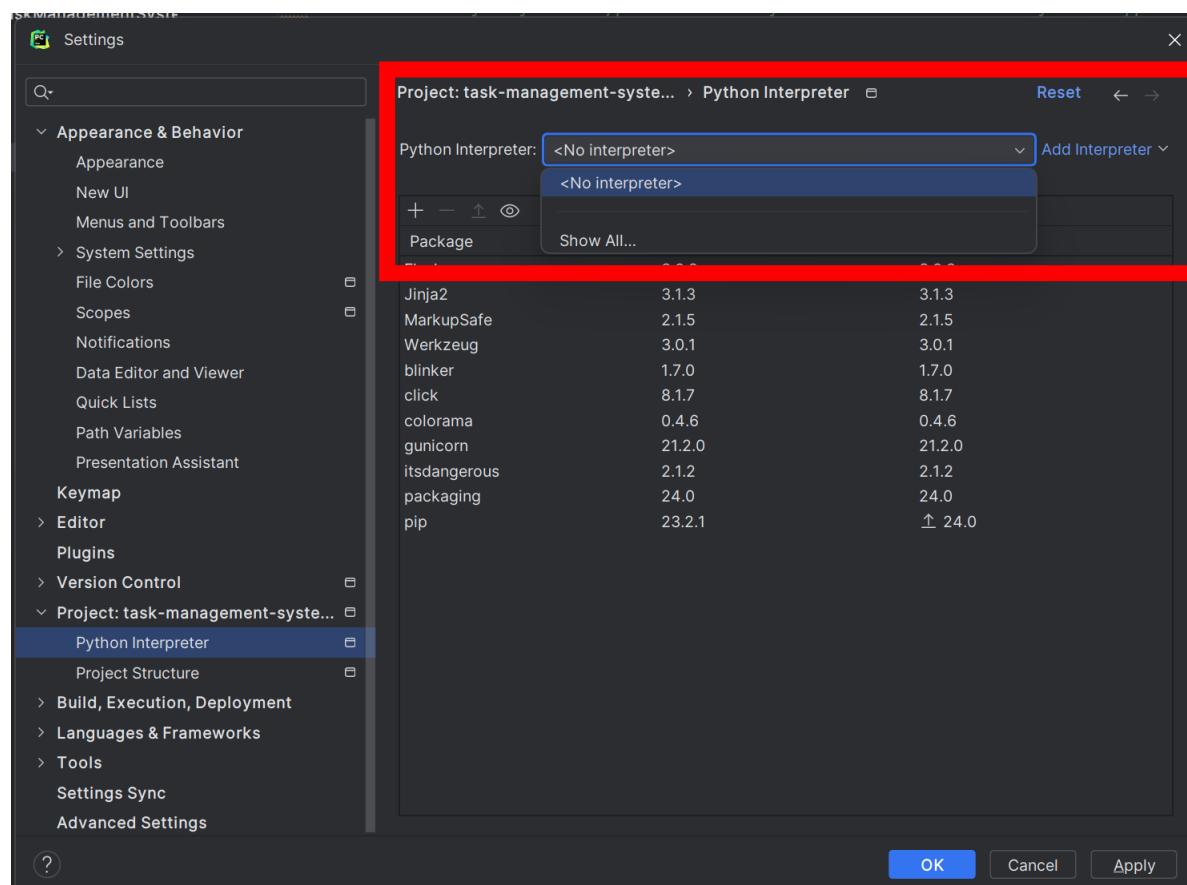


Figure 25 - Setting up the Python Interpreter.

e. Then select "Add Local Interpreter" from the "Add Interpreter" drop-down menu.

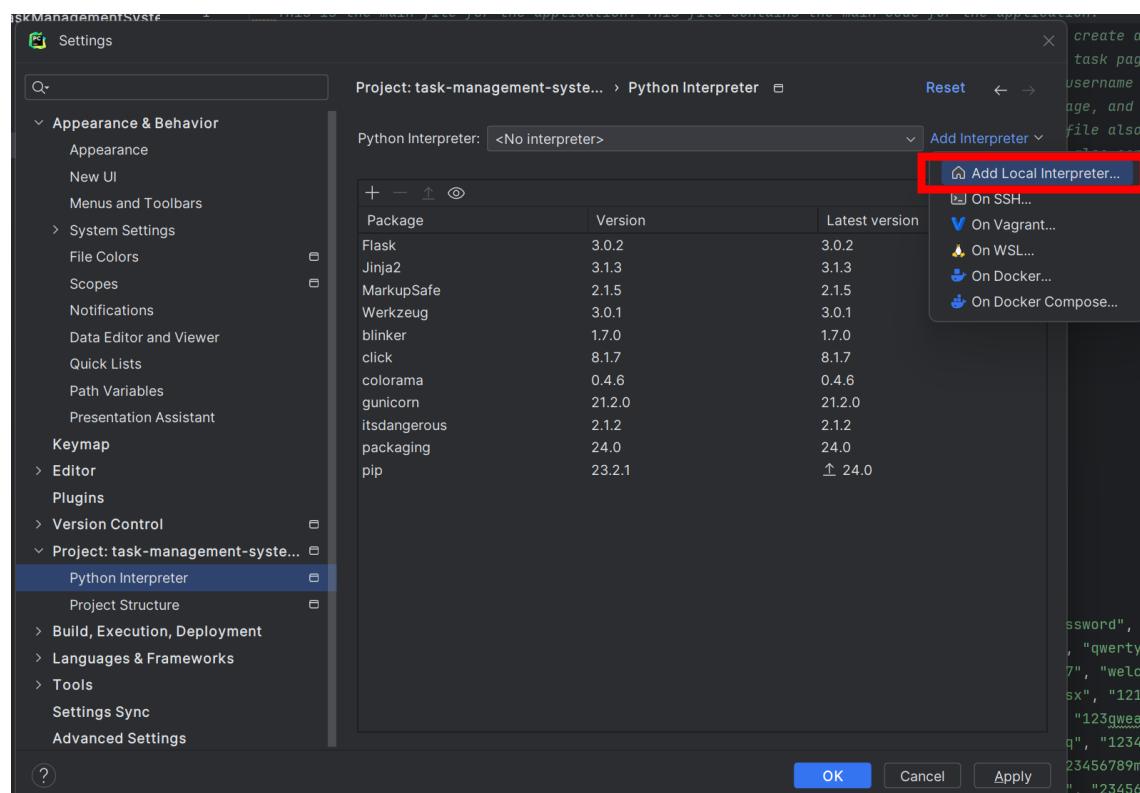


Figure 26 - Setting up the Python Interpreter.

- f. Select “Environment: New” radio button and PyCharm will automatically create a new. venv directory for the Python Virtual Environment. Make sure the Base Interpreter is set to the location path of the Python3.12 interpreter installed on the computer. Then click “OK”.

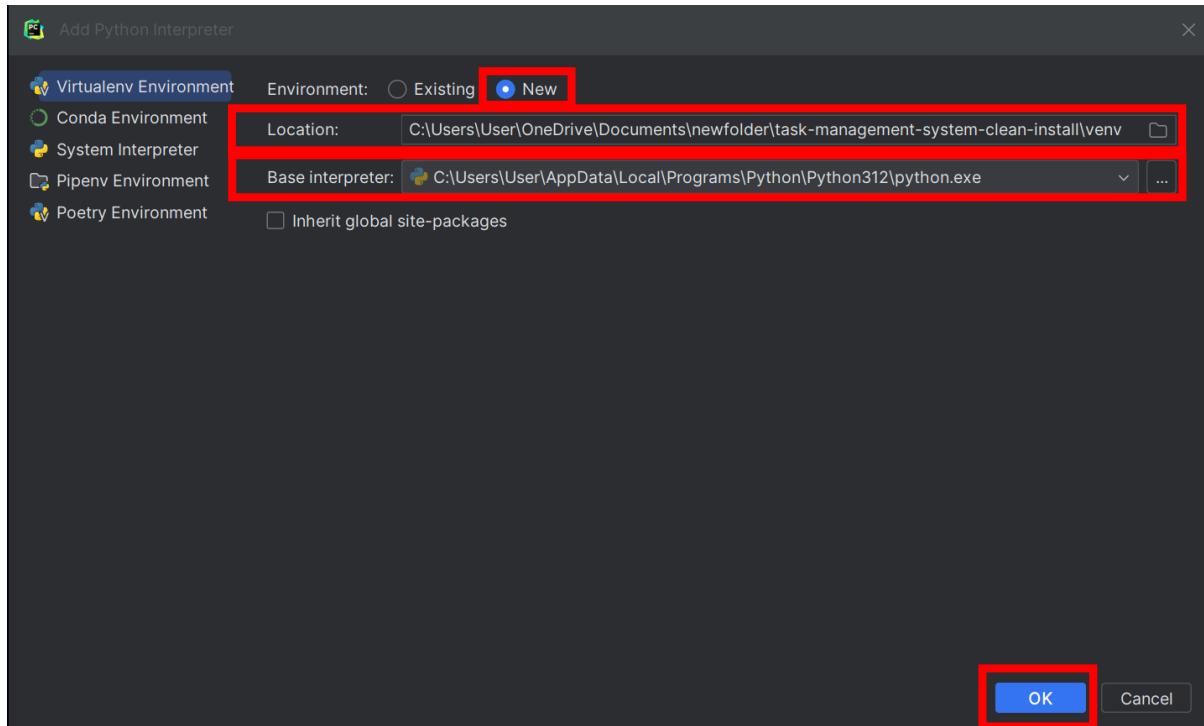


Figure 27 - Select New Environment option, Location: Path to virtual environment (auto generated), Base Interpreter: Path to Python 3.12 installation.

- g. Wait for the virtual environment to be created then click OK again.
 h. Click the link to use the Python Interpreter you have created in the new. venv directory.

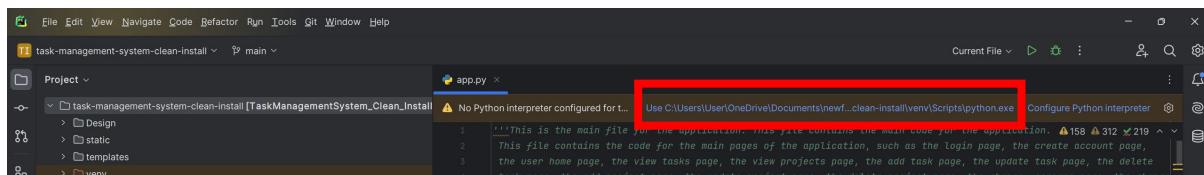


Figure 28 - Click the link to use the Python Interpreter you have created in the new. venv directory.

3. Set up Flask configuration.
 a. Select “Current File” then “Edit Configurations...”

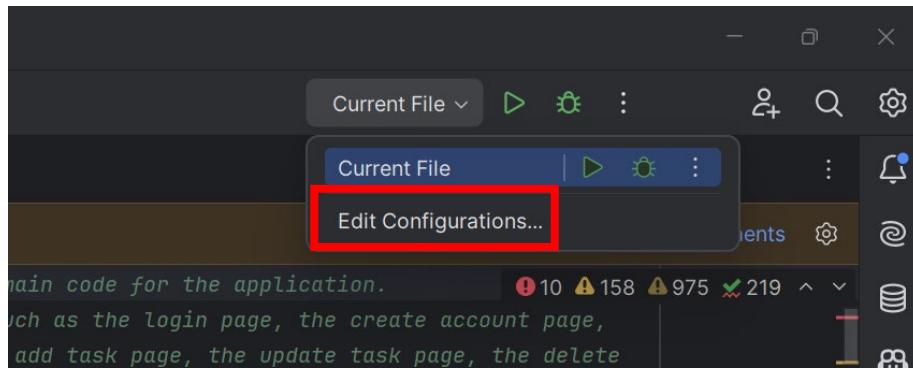


Figure 29 - Select Edit Configurations...

- Click "Add new run configuration..." and select "Flask Server" from the drop-down menu.

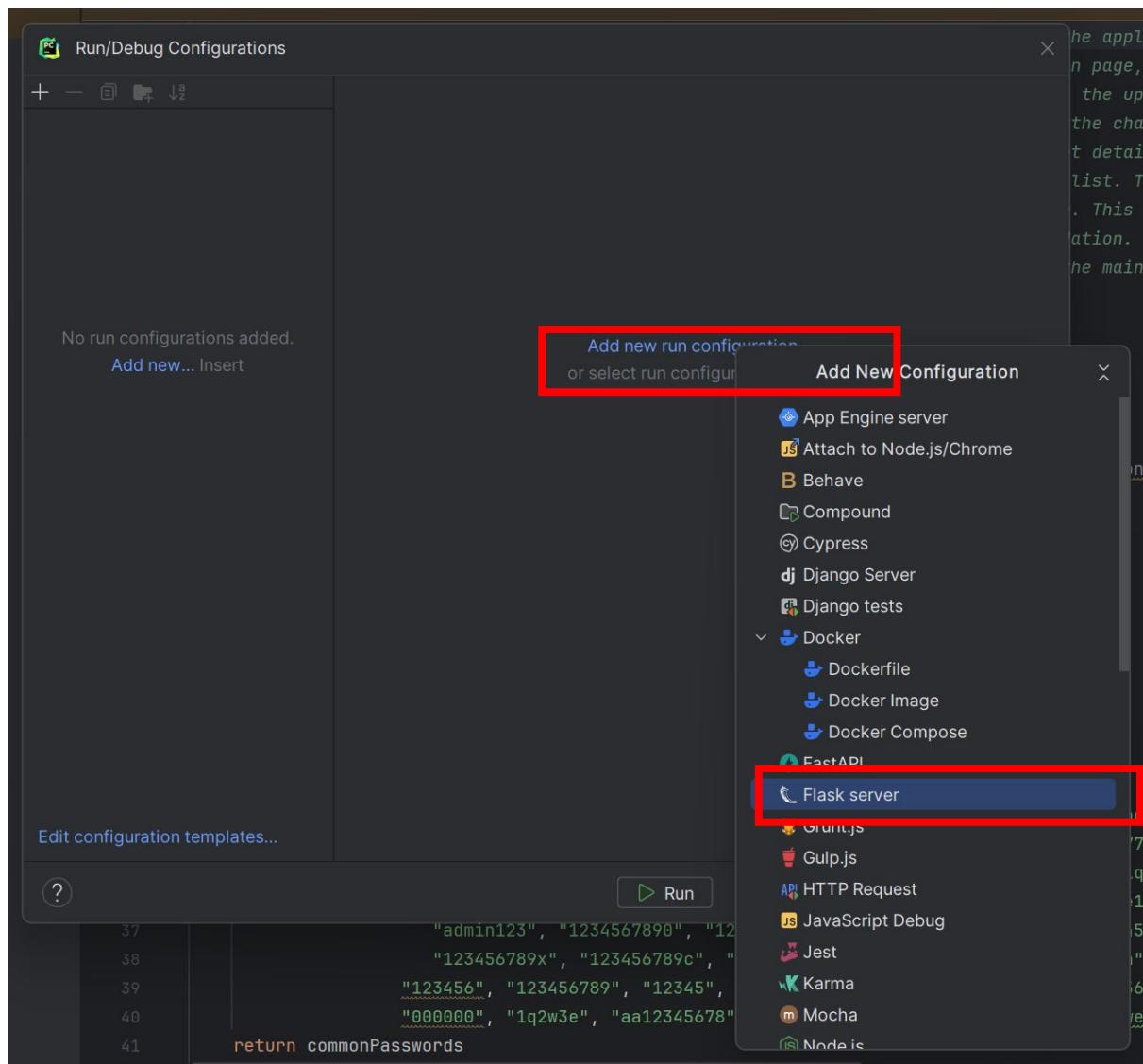


Figure 30 - Select "Add new run configuration" then "Flask server".

- Set the target to "App.py", Set "FLASK_ENV:" to "development" and enable "FLASK_DEBUG". Make sure the Python interpreter is selected in "Python Interpreter". Then click "OK".

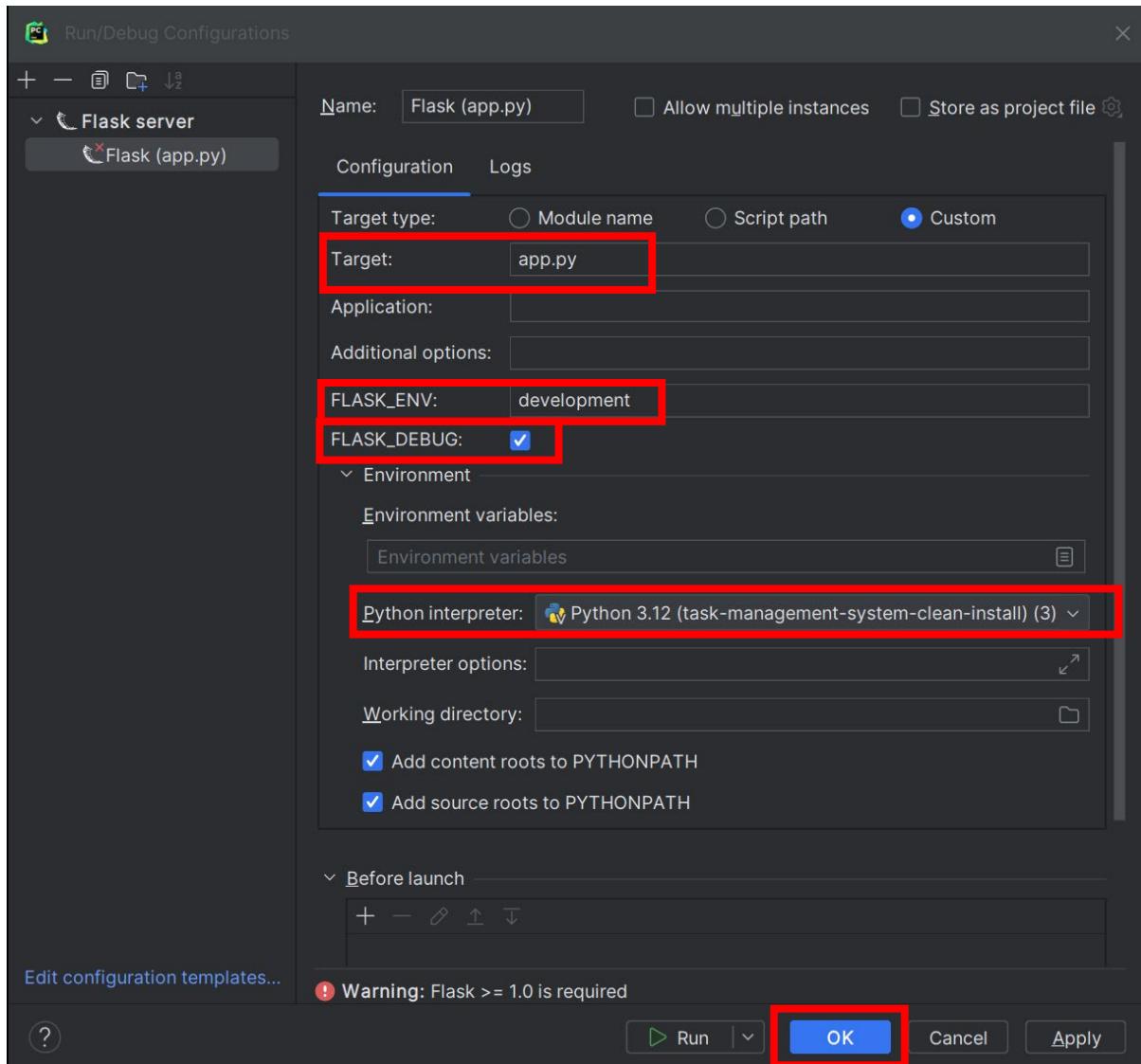


Figure 31 - Set Target to "app.py", FLASK_ENV: to "development", enable FLASK_DEBUG; set Python Interpreter to Python 3.12 (task-management-system(-clean-install)). Click OK.

- d. Install any missing requirements if you get a message that they are missing by clicking "Install requirements".

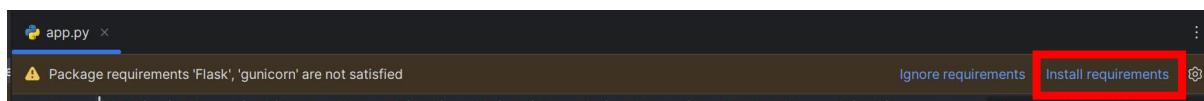


Figure 32 - Install requirements if prompted to do so.

- e. If you had to install any missing requirements, make sure both requirements are selected and click "Install". Wait for requirements to install.

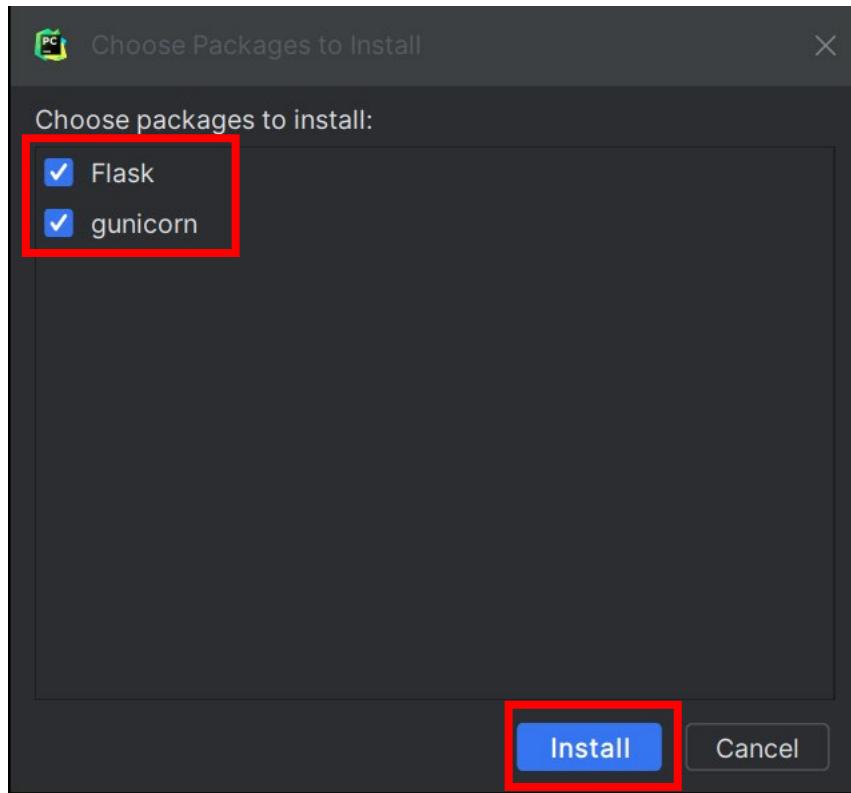


Figure 33 - Make sure "Flask" and "Gunicorn" are selected then click "Install".

i. x

```

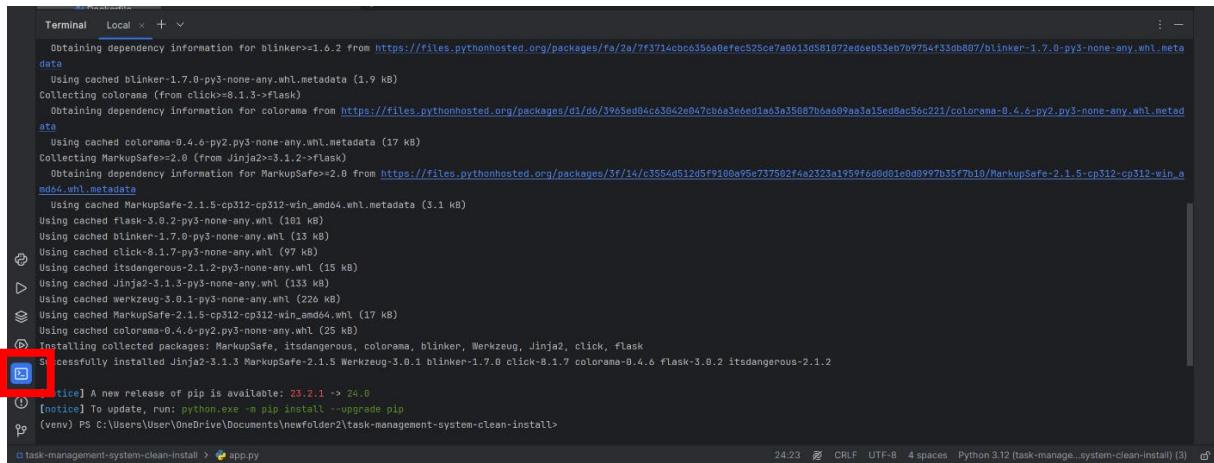
Terminal Local + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(venv) PS C:\Users\User\OneDrive\Documents\newFolder2\task-management-system-clean-install> pip install flask
Collecting flask
  Obtaining dependency information for flask from https://files.pythonhosted.org/packages/93/a6/ae90bfe0eb9e8b15d36cd7d03c8ca86a03968a07f27ce224fb4f766acb23/\*flask-3.0.2-py3-none-any.whl.metadata
    Using cached flask-3.0.2-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug<3.0.0 (from flask)
  Obtaining dependency information for Werkzeug<3.0.0 from https://files.pythonhosted.org/packages/c3/fc/254c3e9b5feb89ff5b9076a23218daefbc9c96ac5941e980b71206e6313b/werkzeug-3.0.1-py3-none-any.whl.metadata
    Using cached werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Obtaining dependency information for Jinja2>=3.1.2 from https://files.pythonhosted.org/packages/30/0d/6de0be2d02603eb56e72997708809e8a5b0fbfee080735109b40a3564843/Jinja2-3.1.3-py3-none-any.whl.metadata
    Using cached Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)
Collecting itsdangerous>=2.1.2 (from flask)
  Obtaining dependency information for itsdangerous>=2.1.2 from https://files.pythonhosted.org/packages/68/5f/447e04e628f47465eeah35b5d408b7ebaasee207f48b713cc5a7267a30ae/itsdangerous-2.1.2-py3-none-any.whl.metadata
    Using cached itsdangerous-2.1.2-py3-none-any.whl.metadata (2.9 kB)
Collecting click>=8.1.3 (from flask)
  Obtaining dependency information for click>=8.1.3 from https://files.pythonhosted.org/packages/00/2e/d53fa4befbf2cfa713304affc7ca780ce4fc1fd8710527771b58311a3229/click-8.1.7-py3-none-any.whl.metadata
    Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from flask)

```

Figure 34 - "pip install flask" in PyCharm console.

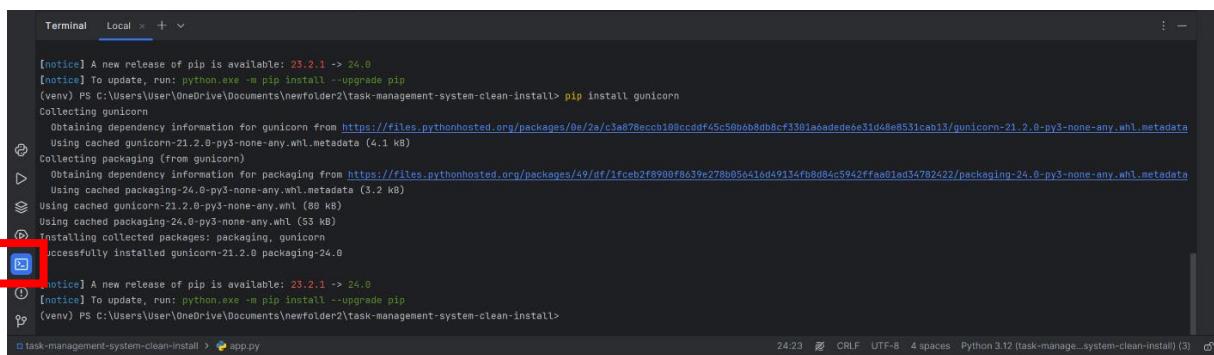


```

Terminal Local + v
Obtaining dependency information for blinker>=1.6.2 from https://files.pythonhosted.org/packages/fa/2a/7f3714cbbc35a0defec525ce7a6613d581072ed6eb53eb7b9754f33db807/blinker-1.7.0-py3-none-any.whl.metadata
Using cached blinker-1.7.0-py3-none-any.whl.metadata (1.9 KB)
Collecting colorama (from click>=8.1.3->flask)
  Obtaining dependency information for colorama from https://files.pythonhosted.org/packages/d1/d6/3965ed84c83842e047cb6a3e6ed1a63a35087b6a609aa3a15ed8ac5dc221/colorama-0.4.6-py2.py3-none-any.whl.metadata
    Using cached colorama-0.4.6-py2.py3-none-any.whl.metadata (17 KB)
  Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
    Obtaining dependency information for MarkupSafe>=2.0 from https://files.pythonhosted.org/packages/3f/14/c3554d512d5f9100e95e737502f4a2323a1959f6d0d01e0d0997b35f7b10/MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata
      Using cached MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata (3.1 kB)
    Using cached Flask-3.0.2-py3-none-any.whl (101 kB)
    Using cached blinker-1.7.0-py3-none-any.whl (19 kB)
    Using cached click-8.1.7-py3-none-any.whl (97 kB)
    Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
  ▷ Using cached Jinja2-3.1.3-py3-none-any.whl (133 kB)
  ▷ Using cached werkzeug-3.0.1-py3-none-any.whl (226 kB)
  ▷ Using cached MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl (17 kB)
  ▷ Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
  ⚠️ Installing collected packages: MarkupSafe, itsdangerous, colorama, blinker, Werkzeug, Jinja2, click, flask
    Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 colorama-0.4.6 flask-3.0.2 itsdangerous-2.1.2
  [!] [notice] A new release of pip is available: 23.2.1 -> 24.0
  [!] [notice] To update, run: python.exe -m pip install --upgrade pip
  [?] (venv) PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install>

```

Figure 35 - "pip install flask" in PyCharm console.



```

Terminal Local + v
[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
(venv) PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install> pip install gunicorn
Collecting gunicorn
  Obtaining dependency information for gunicorn from https://files.pythonhosted.org/packages/0e/2a/c3a878eccb108ccdf4c5c50n6bb8cf3301a5adde6e31d48e8531cab15/gunicorn-21.2.0-py3-none-any.whl.metadata
    Using cached gunicorn-21.2.0-py3-none-any.whl.metadata (4.1 kB)
  ▷ Collecting packaging (from gunicorn)
  ▷ Obtaining dependency information for packaging from https://files.pythonhosted.org/packages/49/df/1fce2b78900f8639e278b05641od49134fb8d84c5942ffea01ad34782422/packaging-24.0-py3-none-any.whl.metadata
    Using cached packaging-24.0-py3-none-any.whl.metadata (2.2 kB)
  ▷ Using cached gunicorn-21.2.0-py3-none-any.whl (88 kB)
  ▷ Using cached packaging-24.0-py3-none-any.whl (53 kB)
  ⚠️ Installing collected packages: packaging, gunicorn
    Successfully installed gunicorn-21.2.0 packaging-24.0
  [!] [notice] A new release of pip is available: 23.2.1 -> 24.0
  [!] [notice] To update, run: python.exe -m pip install --upgrade pip
  [?] (venv) PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install>

```

Figure 36 - "pip install gunicorn" in PyCharm console.

Setting up the PyCharm Environment (Linux)

1. Delete the venv directory and confirm deletion.

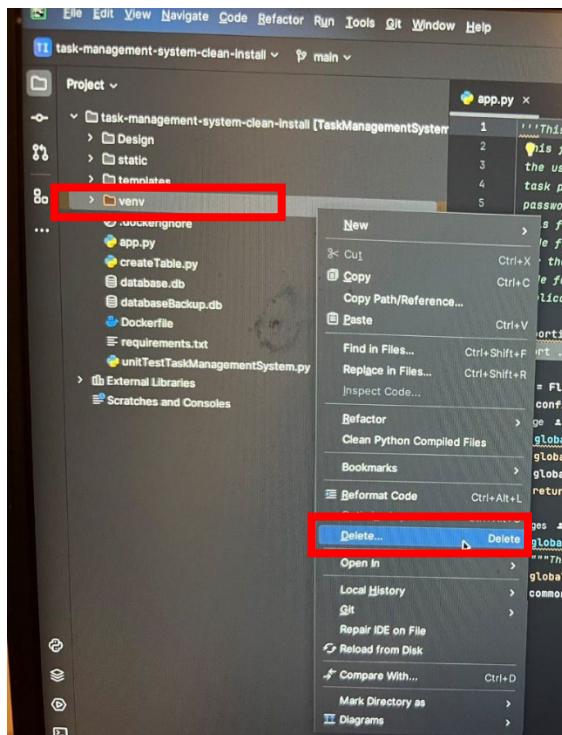


Figure 37 - Deleting the. venv or venv folder

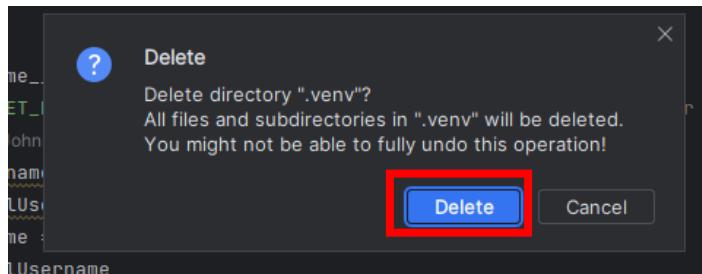


Figure 38 - Click Delete

2. Select the Python Interpreter (Method 1)

- Click link to “Configure Python Interpreter” then “Add New Interpreter” then “Add Local Interpreter”.

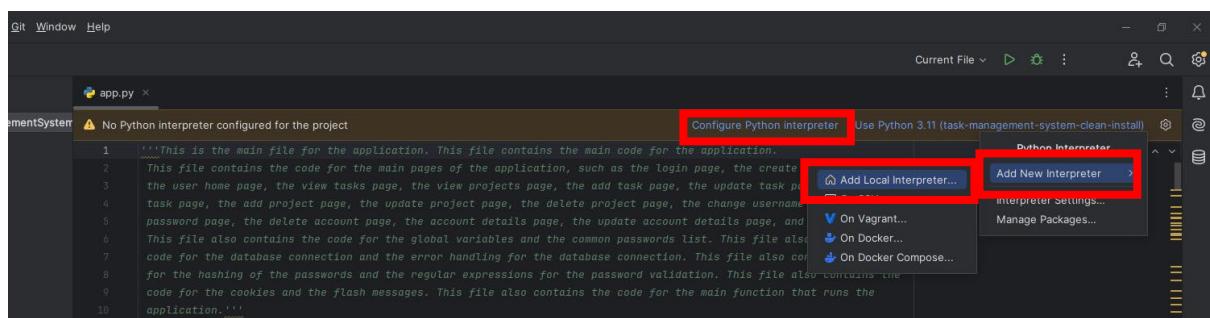


Figure 39 - Select "Configure Python Interpreter" > "Add New Interpreter" > "Add Local Interpreter..."

- Select “New” for Environment, PyCharm will automatically select the task-management-system or task-management-system-clean-install directory and create a new .venv folder containing the Python virtual environment. Make sure that the location of the Base Interpreter is the location where the Python 3.12 is installed on your computer. Then click “OK”.

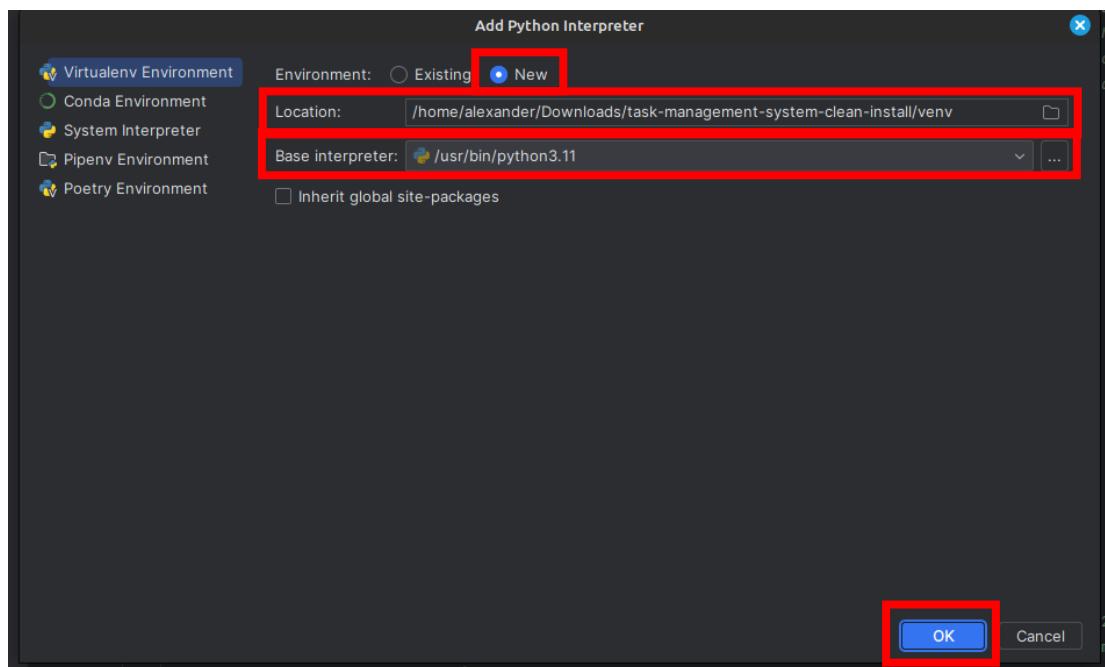


Figure 40 - Adding a Python Interpreter.

- c. Click on the link to “Install Requirements” when it appears. If this doesn’t appear, enter the PIP commands `pip install flask` and `pip install gunicorn` into the PyCharm Terminal and wait for the modules to install as an alternative method.

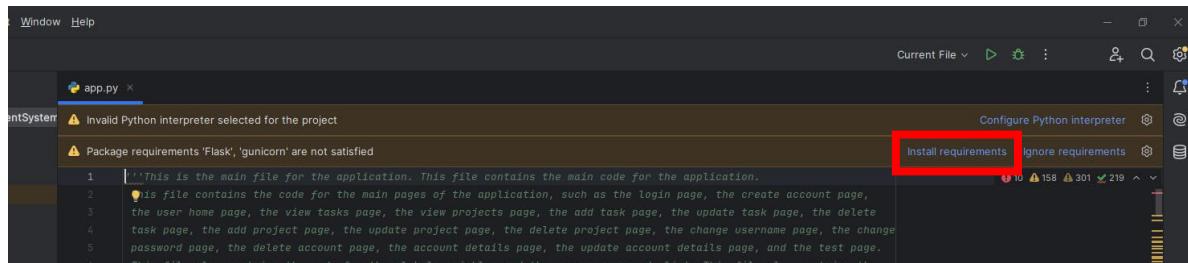


Figure 41 - Select "Install requirements".

- d. Make sure “Flask” and “Gunicorn” checkboxes are selected then click “Install”.

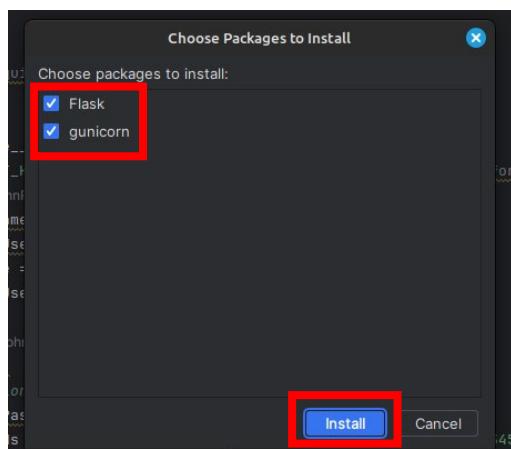


Figure 42 - Make sure "Flask" and "Gunicorn" are selected then click "Install".

The screenshot shows the PyCharm IDE interface with a terminal window open. The terminal output is as follows:

```

File Edit View Navigate Code Refactor Run Tools Git Window Help
task-management-system-clean-install main
Project Terminal Local + 
Silence deprecation warnings for upcoming unsupported Python versions.
--use-feature <feature> Enable new functionality, that may be backward incompatible.
--use-deprecated <feature> Enable deprecated functionality, that will be removed in the future.
(venv) pip install flask
Collecting flask
  Using cached Flask-3.0.2-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Using cached werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Using cached Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)
Collecting itsdangerous>=2.1.2 (from flask)
  Using cached itsdangerous-2.1.2-py3-none-any.whl.metadata (2.9 kB)
Collecting click>=8.1.3 (from flask)
  Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from flask)
  Using cached blinker-1.7.0-py3-none-any.whl.metadata (1.9 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
  Using cached MarkupSafe-2.1.5-cp311-cp311-manylinux_2_17_x86_64_manylinux2014_x86_64.whl.metadata (3.0 kB)
Using cached flask-3.0.2-py3-none-any.whl (101 kB)
Using cached blinker-1.7.0-py3-none-any.whl (13 kB)
Using cached click-8.1.7-py3-none-any.whl (97 kB)
Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Using cached Jinja2-3.1.3-py3-none-any.whl (133 kB)
Using cached werkzeug-3.0.1-py3-none-any.whl (226 kB)
Using cached MarkupSafe-2.1.5-cp311-cp311-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (28 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, flask
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 itsdangerous-2.1.2
(venv) pip install gunicorn
Collecting gunicorn
  Using cached gunicorn-21.2.0-py3-none-any.whl.metadata (4.1 kB)
Requirement already satisfied: packaging in /app/lib/python3.11/site-packages (from gunicorn) (23.2)
Using cached gunicorn-21.2.0-py3-none-any.whl (80 kB)
Installing collected packages: gunicorn
Successfully installed gunicorn-21.2.0
(venv) 

```

Figure 43 - enter "pip install flask" then "pip install gunicorn" into the PyCharm terminal.

2. Select the Python Interpreter (Method 2)

- After deleting the venv folder, a link to “Create a virtual environment using requirements.txt” will appear. Click this link.

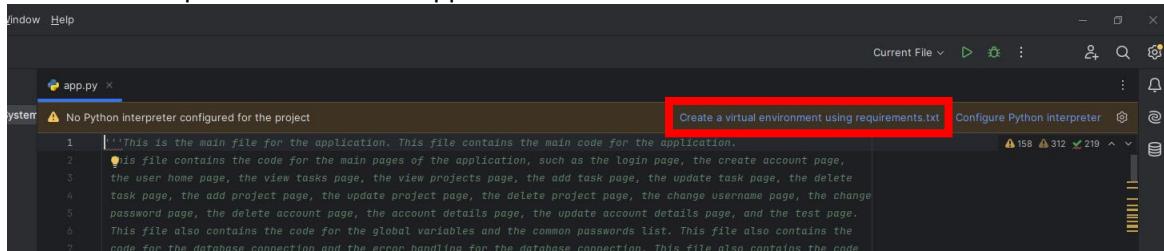


Figure 44 - Click "Create a virtual environment using requirements.txt".

- The Location, Base Interpreter and Dependencies should automatically be set up in the Creating Virtual Environment popup window. Make sure the location is set to the path of the task-management-system/venv or task-management-system-clean-install/venv, the Base Interpreter path is set to the location of the Python 3.12 interpreter install location on your computer and the Dependencies path is set to task-management-system/requirements.txt or task-management-system-clean-install/requirements.txt. Then click "OK", Wait for PyCharm to create the virtual environment and install the dependencies and requirements.

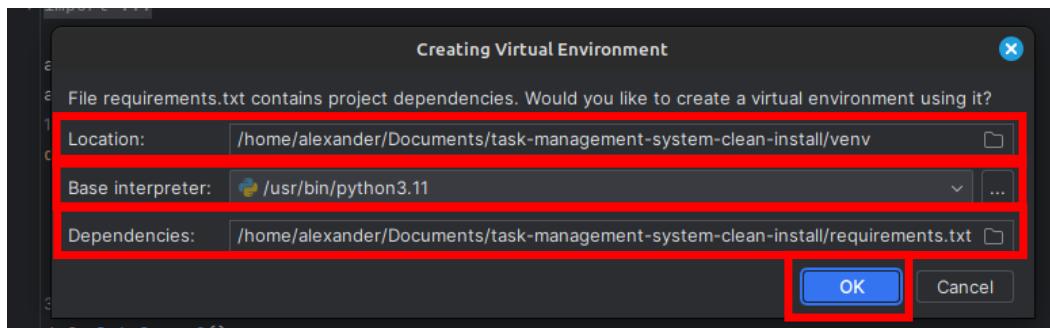


Figure 45 - Location is automatically chosen as the Task Management System's source code root directory. Set the Base interpreter as the location of the Python 3.12 install location and set dependencies to the location of "requirements.txt" in Task Management System's root directory. This should be done automatically. Click "OK".

Setting up the PyCharm environment (macOS)

1. Delete the venv directory then confirm the delete by clicking "OK".

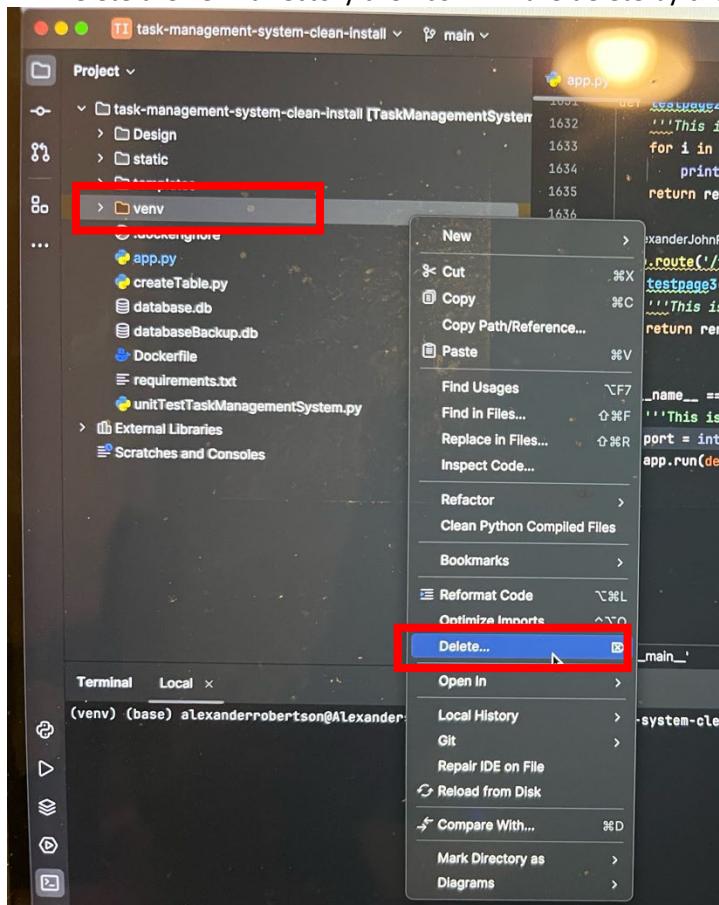


Figure 46 - Deleting the .venv or venv folder.

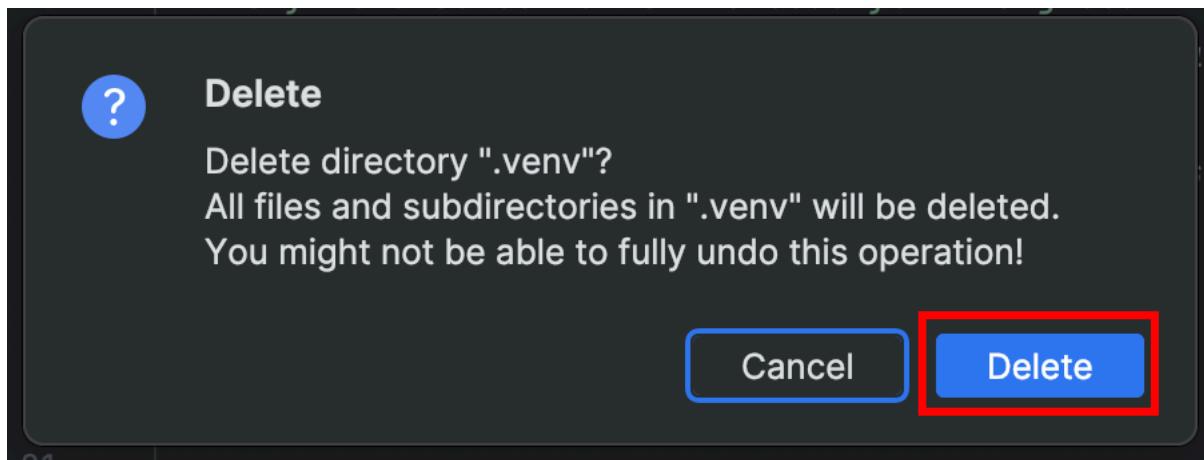


Figure 47 - Click Delete

2. Method 1:

- a. select “Configure Python Interpreter” > “Add New Interpreter” > “Add Local Interpreter”.

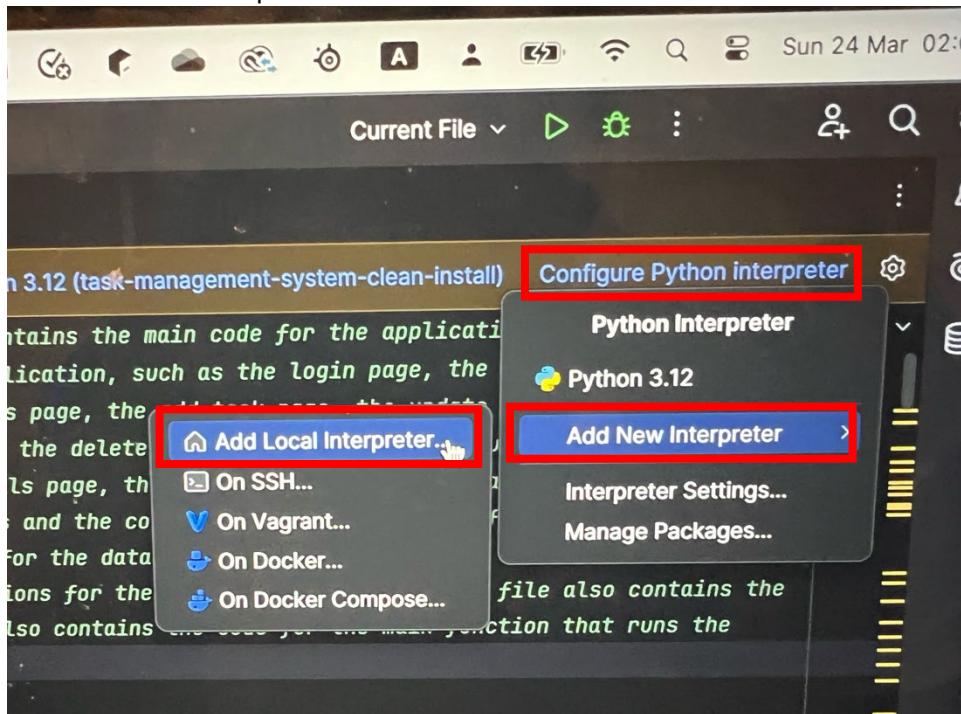


Figure 48 - Select "Configure Python Interpreter" > "Add New Interpreter" > "Add Local Interpreter...".

- b. Select Environment as “New” (This should be automatically selected). The environment location path will be automatically set to the location of the task-management-system/venv or task-management-system-clean-install/venv directory and The Base Interpreter path should be to the location where Python 3.12 is installed on your Mac. Then click “OK”. Wait for PyCharm to create the virtual environment.

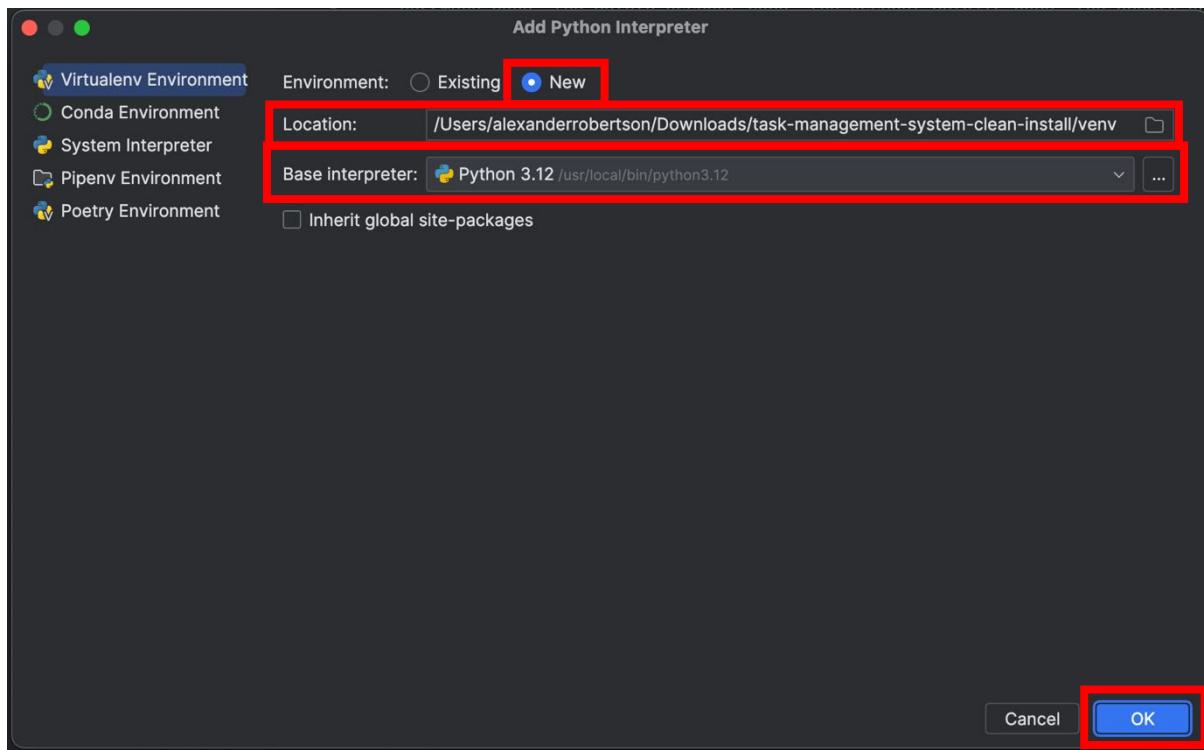


Figure 49 - Adding a Python Interpreter.

- c. Select “Use Python 3.12 (task-management-system-clean-install)” or “Use Python 3.12 (task-management-system)”.

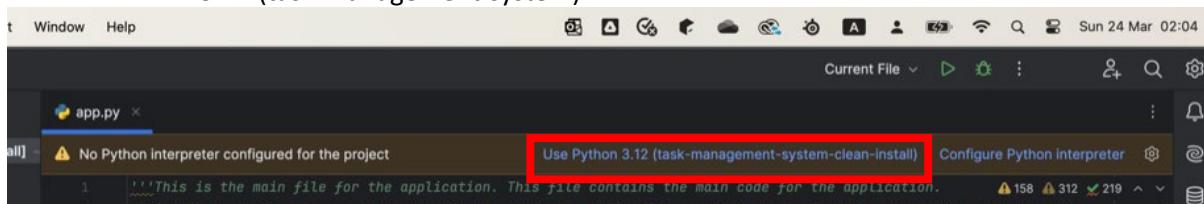


Figure 50 - Select "Use Python 3.12 (task-management-system-clean-install)" or "Use Python 3.12 (task-management-system)".

- d. Select “Install requirements”

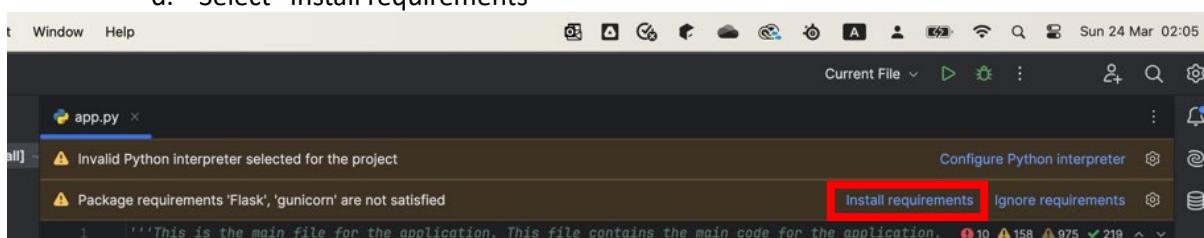


Figure 51 - Select "Install requirements".

- e. Make sure “Flask” and “Gunicorn” checkboxes are both selected then click “Install Requirements”.

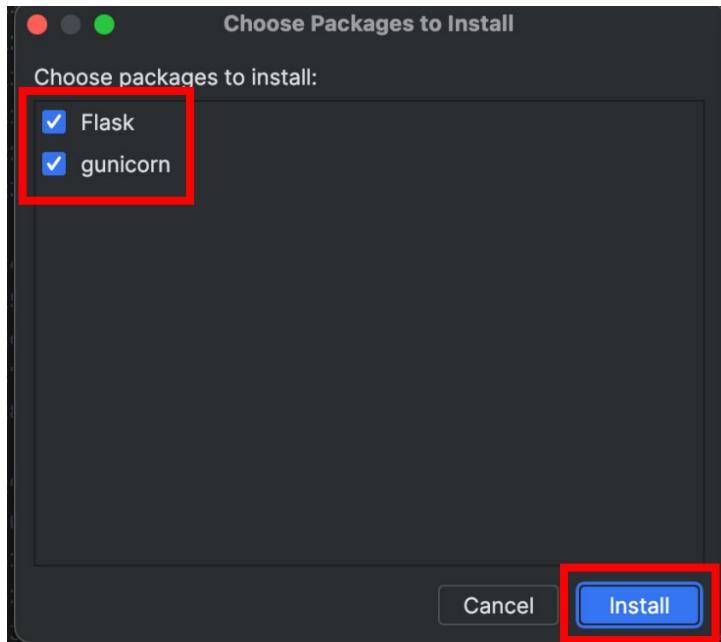


Figure 52 - Make sure "Flask" and "Gunicorn" are selected then click "Install".

- f. Wait for requirements to install.
- g. Alternatively, instead of clicking "Install Requirements" you can enter "pip install flask" and "pip install gunicorn" into the PyCharm terminal.

```
(venv) (base) 14:00:00@ MacBook-Pro task-management-system-clean-install % pip install flask
Collecting flask
  Obtaining dependency information for flask from https://files.pythonhosted.org/packages/93/a6/aa98bfe0eb9b8b15d36cdfd03c8ca86a...
    Using cached flask-3.0.2-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Obtaining dependency information for Werkzeug>=3.0.0 from https://files.pythonhosted.org/packages/c3/fc/254c3e9b5feb89ff5b9076...
    Using cached werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Obtaining dependency information for Jinja2>=3.1.2 from https://files.pythonhosted.org/packages/30/6d/6de6be2d02603ab56e729977...
    Using cached Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)
Collecting itsdangerous>=2.1.2 (from flask)
  Obtaining dependency information for itsdangerous>=2.1.2 from https://files.pythonhosted.org/packages/68/5f/447e04e828f47465ee...
    Using cached itsdangerous-2.1.2-py3-none-any.whl.metadata (2.9 kB)
Collecting click>=8.1.3 (from flask)
  Obtaining dependency information for click>=8.1.3 from https://files.pythonhosted.org/packages/00/2e/d53fa4befbf2cfa713304affc...
    Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from flask)
  Obtaining dependency information for blinker>=1.6.2 from https://files.pythonhosted.org/packages/fa/2a/7f3714cbc6356a0efec525c...
    Using cached blinker-1.7.0-py3-none-any.whl.metadata (1.9 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
  Obtaining dependency information for MarkupSafe>=2.0 from https://files.pythonhosted.org/packages/53/bd/583bf3e4c8d6a321938c13...
    Using cached MarkupSafe-2.1.5-cp312-cp312-macosx_10_9_universal2.whl.metadata (3.0 kB)
Using cached flask-3.0.2-py3-none-any.whl (101 kB)
Using cached blinker-1.7.0-py3-none-any.whl (13 kB)
Using cached click-8.1.7-py3-none-any.whl (97 kB)
Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Using cached Jinja2-3.1.3-py3-none-any.whl (133 kB)
Using cached werkzeug-3.0.1-py3-none-any.whl (226 kB)
Using cached MarkupSafe-2.1.5-cp312-cp312-macosx_10_9_universal2.whl (18 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, flask
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 itsdangerous-2.1.2
[!] [notice] A new release of pip is available: 23.2.1 -> 24.0
[!] [notice] To update, run: pip install --upgrade pip
(venv) (base) 14:00:10@ MacBook-Pro task-management-system-clean-install %
task-management-system-clean-install
```

Figure 53- "pip install flask" in PyCharm console.

```
(venv) (base) -MacBook-Pro task-management-system-clean-install % pip install gunicorn
Collecting gunicorn
  Obtaining dependency information for gunicorn from https://files.pythonhosted.org/packages/0e/2a/c3a878eccb100ccddf45c50b6b8db...
    Using cached gunicorn-21.2.0-py3-none-any.whl.metadata (4.1 kB)
Collecting packaging (from gunicorn)
  Obtaining dependency information for packaging from https://files.pythonhosted.org/packages/49/df/1fcebf8900f8639e278b056416d...
    Using cached packaging-24.0-py3-none-any.whl.metadata (3.2 kB)
    Using cached gunicorn-21.2.0-py3-none-any.whl (80 kB)
    Using cached packaging-24.0-py3-none-any.whl (53 kB)
  Installing collected packages: packaging, gunicorn
    Successfully installed gunicorn-21.2.0 packaging-24.0

  [notice] A new release of pip is available: 23.2.1 -> 24.0
  [notice] To update, run: pip install --upgrade pip
(venv) (base) -MacBook-Pro task-management-system-clean-install %
task-management-system-clean-install
```

Figure 54 – “pip install gunicorn” into the PyCharm terminal.

3. Method 2:

- Wait for “No Python interpreter configured for this project” error to appear above the code. Then click “Use <path/to/task-management-system-clean-install/.venv/Scripts/python.exe>” or “Use <path/to/task-management-system/.venv/Scripts/python.exe>”

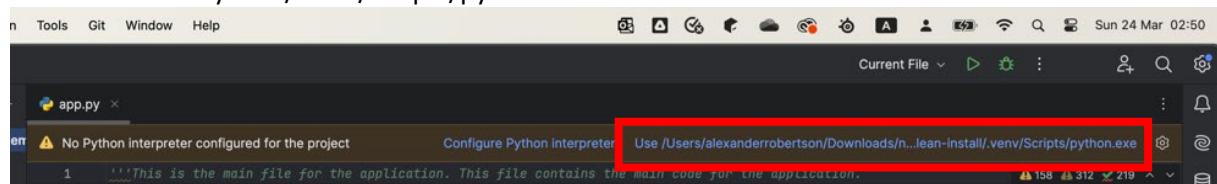


Figure 55 - Select "Use <path/to/task-management-system-clean-install/.venv/scripts/python.exe>" or "Use <path/to/task-management-system/.venv/Scripts/python.exe>".

- Then click “Create a virtual environment using requirements.txt”.

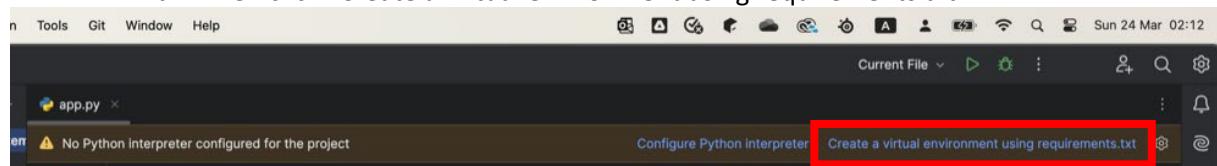


Figure 56 - Click "Create a virtual environment using requirements.txt".

- The Location, Base interpreter and Dependencies paths are automatically generated. The Location should be “</Path/to/task-management-system-clean-install/venv> or </Path/to/task-management-system/venv>”, the Base interpreter should be the path to the location where the Python 3.12 interpreter is installed on your Mac and Dependencies should be </Path/to/task-management-system-clean-install/requirements.txt> or </Path/to/task-management-system/requirements.txt>. then click “OK” and wait for PyCharm to create the virtual environment.

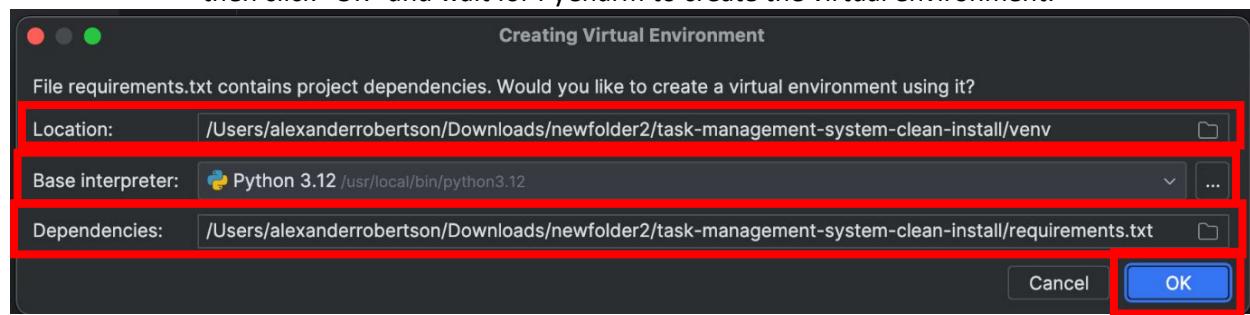


Figure 57 - Location is automatically chosen as the Task Management System's source code root directory. Set the Base interpreter as the location of the Python 3.12 install location and set dependencies to the location of "requirements.txt" in Task Management.

Setting up the Flask Configuration

1. Select “Current File” then “Edit Configurations...”

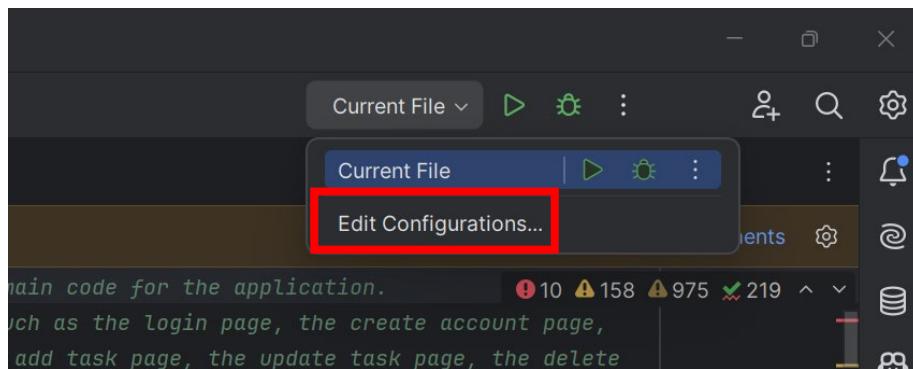


Figure 58 - Select Edit Configurations...

2. Click “Add new run configuration...” and select “Flask Server” from the drop-down menu.

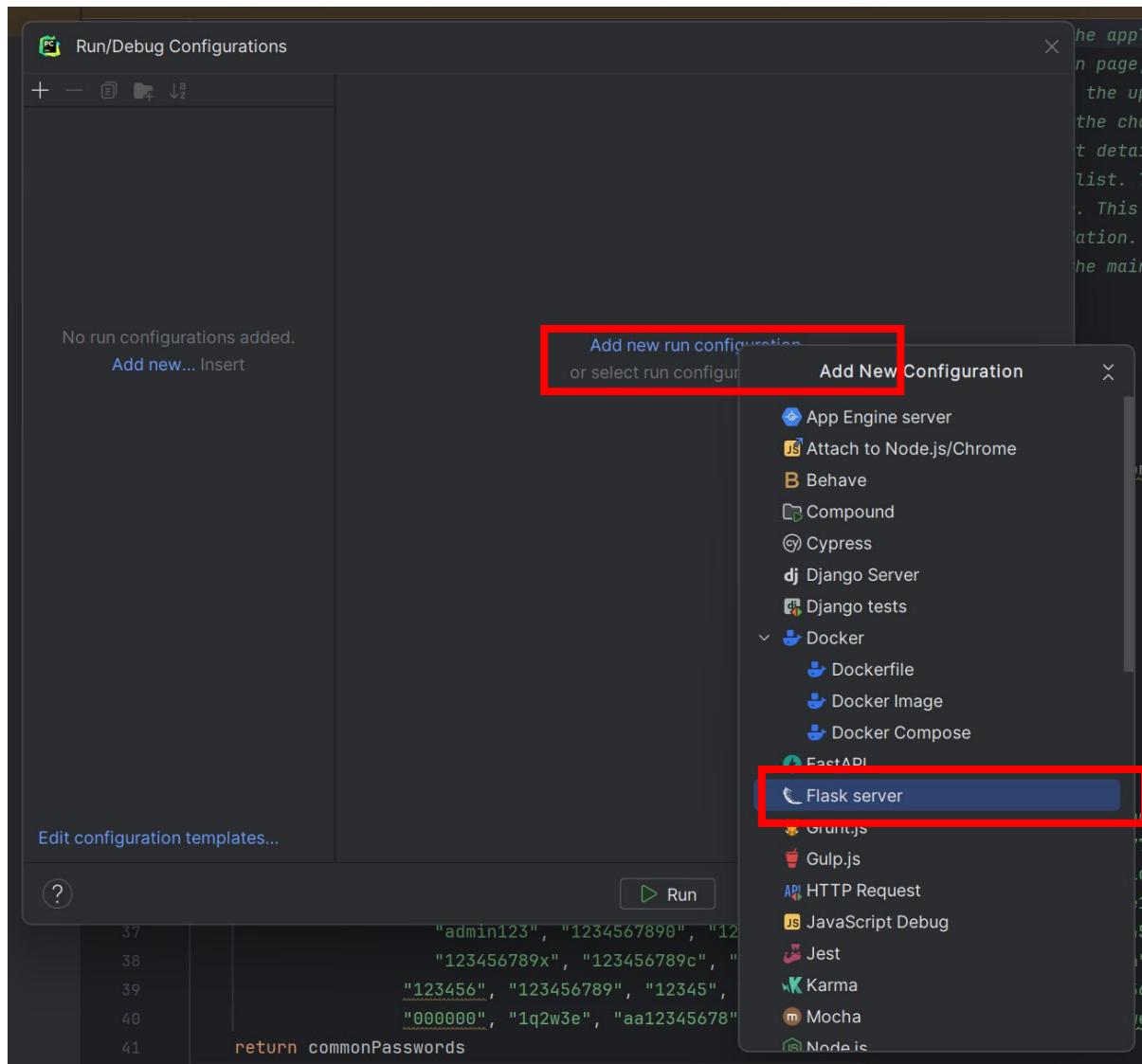


Figure 59 - Select "Add new run configuration" then "Flask server".

3. Set the target to "App.py", Set "FLASK_ENV:" to "development" and enable "FLASK_DEBUG". Make sure the Python interpreter is selected in "Python Interpreter". Then click "OK".

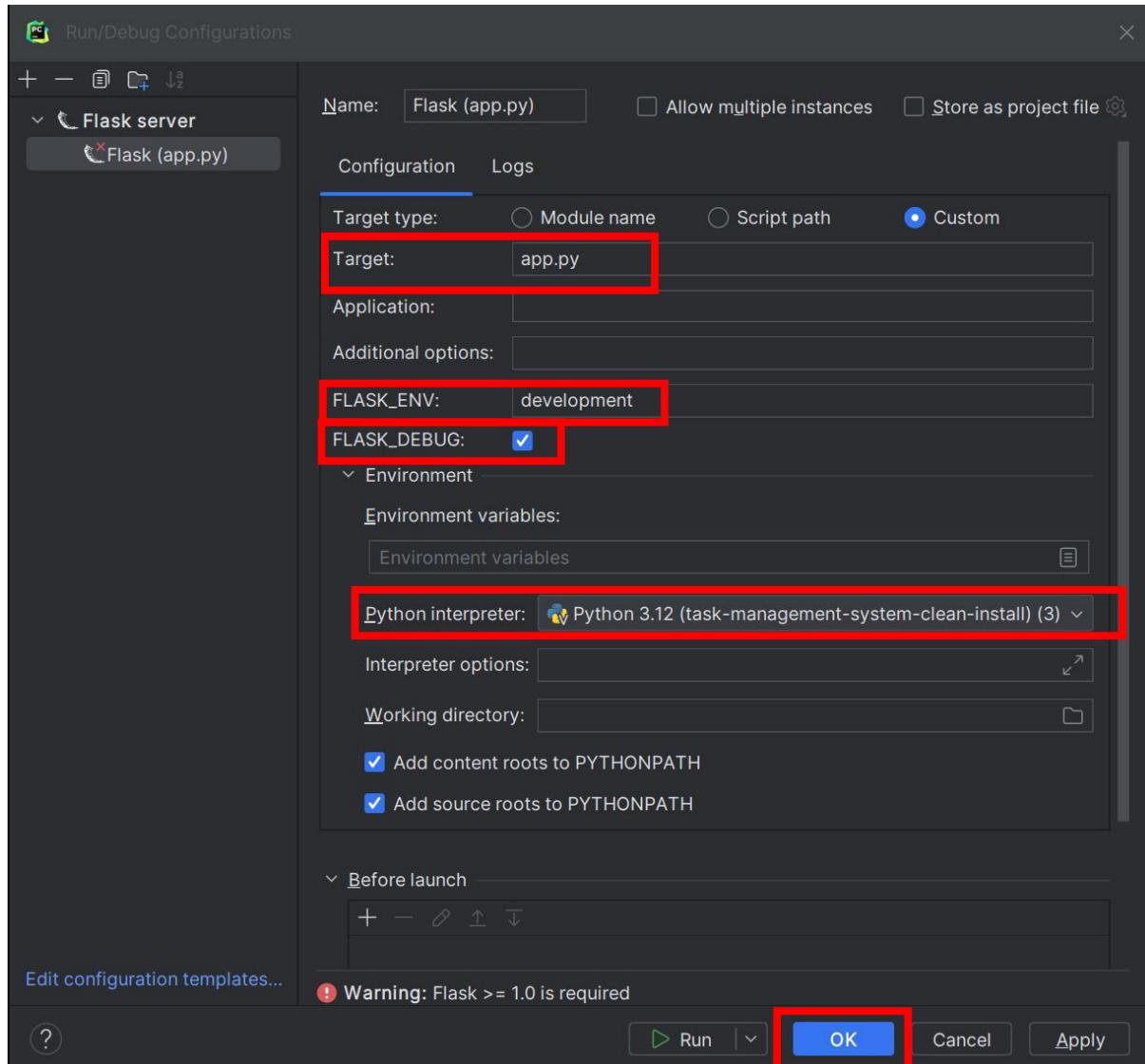


Figure 60 - Set Target to "app.py", FLASK_ENV: to "development", enable FLASK_DEBUG; set Python Interpreter to Python 3.12 (task-management-system(-clean-install)). Click OK.

4. Install any missing requirements if you get a message that they are missing by clicking "Install requirements".

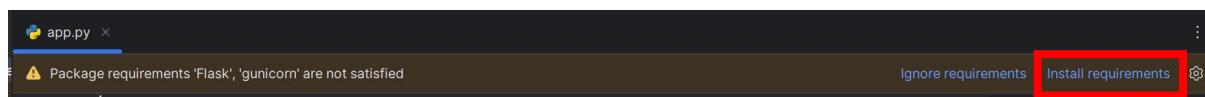


Figure 61 - Install requirements if prompted to do so.

5. If you had to install any missing requirements, make sure both requirements are selected and click "Install". Wait for requirements to install.

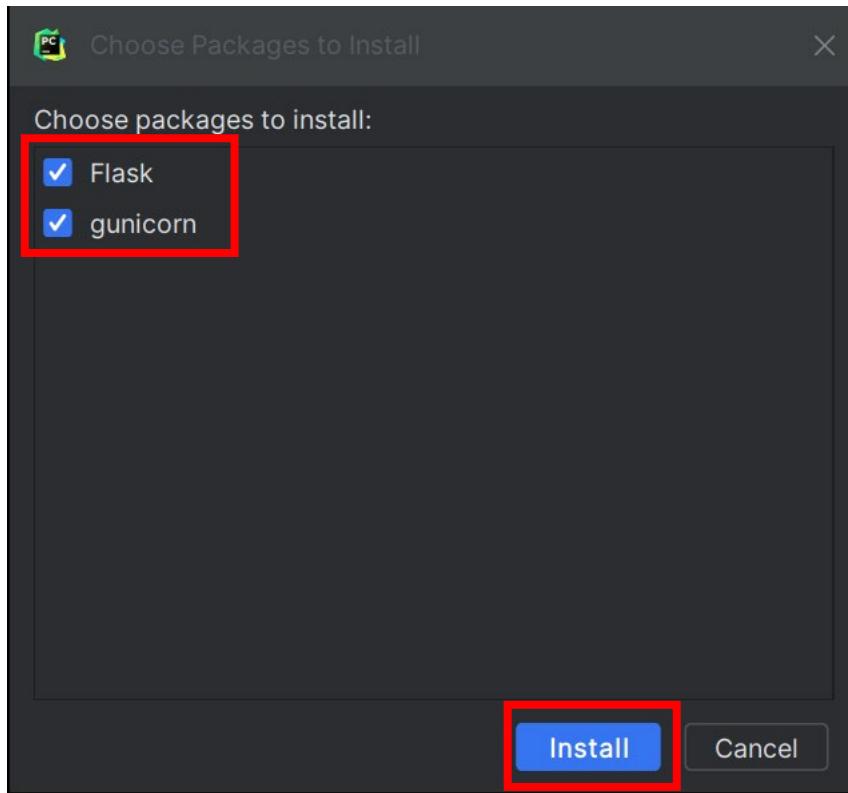


Figure 62 - Make sure "Flask" and "Gunicorn" are selected then click "Install".

6. If the prompt to install the requirements does not appear in the PyCharm IDE, use the PyCharm terminal to install Flask and Gunicorn by entering the following commands: `pip install flask` and then `pip install gunicorn`. Wait for the pip installations to complete.

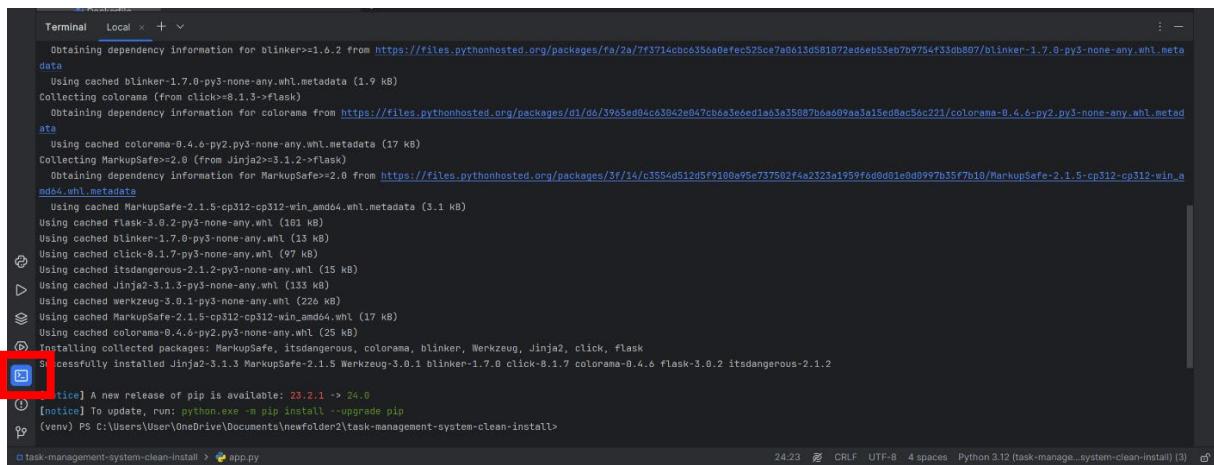
```

Terminal Local + ×
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(venv) PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install> pip install flask
Collecting flask
  Obtaining dependency information for flask from https://files.pythonhosted.org/packages/93/a6/aa98bfe0eb9b8b15d36cdfd03c8ca86a03968a87f27ce224fb4f766acb23/flask-3.0.2-py3-none-any.whl.metadata
    Using cached flask-3.0.2-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Obtaining dependency information for Werkzeug>=3.0.0 from https://files.pythonhosted.org/packages/c3/fc/254c3e9b5feb89ff5b9876a23218daefbc99c96ac5941e980b71206ee6313b/werkzeug-3.0.1-py3-none-any.whl.metadata
    Using cached werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Obtaining dependency information for Jinja2>=3.1.2 from https://files.pythonhosted.org/packages/30/6a/0de6be2d02683ab56e72997708809e8a5b0fbfee080735109b40a3564843/Jinja2-3.1.3-py3-none-any.whl.metadata
    Using cached Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)
Collecting itsdangerous>=2.1.2 (from flask)
  Obtaining dependency information for itsdangerous>=2.1.2 from https://files.pythonhosted.org/packages/68/5f/447e04e828f47465eab35b5d408b7ebaasee207f48b7136c5a7267a30ae/itsdangerous-2.1.2-py3-none-any.whl.metadata
    Using cached itsdangerous-2.1.2-py3-none-any.whl.metadata (2.9 kB)
Collecting click>=8.1.3 (from flask)
  Obtaining dependency information for click>=8.1.3 from https://files.pythonhosted.org/packages/00/2e/d53fa4befbf2cfaf713304effc7ca780ce4fc1fd8710527771b58311a3229/click-8.1.7-py3-none-any.whl.metadata
    Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from flask)
  Obtaining dependency information for blinker>=1.6.2 from https://files.pythonhosted.org/packages/
```

Figure 63 - "pip install flask" in PyCharm console.

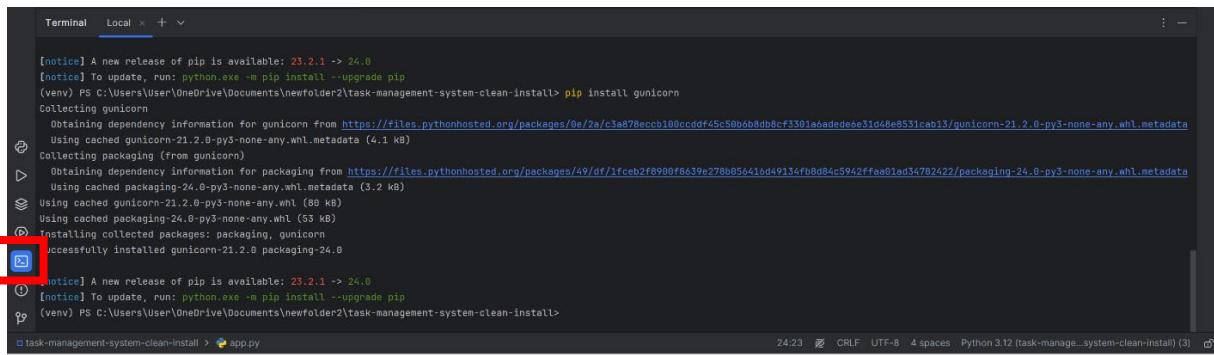


```

Terminal Local + - ⋮
Obtaining dependency information for blinker>=1.6.2 from https://files.pythonhosted.org/packages/fa/7f/3714cbbc356a0efec525ce7a6613d581072ed6eb53eb7b9754f33db807/blinker-1.7.0-py3-none-any.whl.metadata
Using cached blinker-1.7.0-py3-none-any.whl.metadata (1.9 kB)
Collecting colorama (from click>=8.1.3->flask)
  Obtaining dependency information for colorama from https://files.pythonhosted.org/packages/d1/d6/3965ed84c83842e047cb6a3e6ed1a63a35087b6a6d09aa3a15ed8ac58c221/colorama-0.4.6-py2.py3-none-any.whl.metadata
    Using cached colorama-0.4.6-py2.py3-none-any.whl.metadata (17 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
  Obtaining dependency information for MarkupSafe>=2.0 from https://files.pythonhosted.org/packages/3f/14/c3554d512d5f9100e95e737502f4a2323a1959f6d0d01e0d0997b35f7b10/MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata
    Using cached MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata (3.1 kB)
Using cached Flask-3.0.2-py3-none-any.whl (101 kB)
Using cached blinker-1.7.0-py3-none-any.whl (15 kB)
Using cached click-8.1.7-py3-none-any.whl (97 kB)
Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Using cached Jinja2-3.1.3-py3-none-any.whl (135 kB)
Using cached werkzeug-3.0.1-py3-none-any.whl (226 kB)
Using cached MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl (17 kB)
Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, colorama, blinker, Werkzeug, Jinja2, click, flask
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 colorama-0.4.6 flask-3.0.2 itsdangerous-2.1.2
[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
[venv] PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install>

```

Figure 64 - "pip install flask" in PyCharm console.



```

Terminal Local + - ⋮
[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
[venv] PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install> pip install gunicorn
Collecting gunicorn
  Obtaining dependency information for gunicorn from https://files.pythonhosted.org/packages/0e/2a/c3a878eccb108ccdf4c5c50b6b8cf3301a5adde6e31d48e8531ca15/gunicorn-21.2.0-py3-none-any.whl.metadata
    Using cached gunicorn-21.2.0-py3-none-any.whl.metadata (4.1 kB)
Collecting packaging (from gunicorn)
  Obtaining dependency information for packaging from https://files.pythonhosted.org/packages/49/df/1fce2f18900f8639e278b056416d49134fb8d84c5942ffea01ad34782422/packaging-24.0-py3-none-any.whl.metadata
    Using cached packaging-24.0-py3-none-any.whl.metadata (3.2 kB)
Using cached gunicorn-21.2.0-py3-none-any.whl (88 kB)
Using cached packaging-24.0-py3-none-any.whl (53 kB)
Installing collected packages: packaging, gunicorn
Successfully installed gunicorn-21.2.0 packaging-24.0
[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
[venv] PS C:\Users\User\OneDrive\Documents\newfolder2\task-management-system-clean-install>

```

Figure 65 - "pip install gunicorn" in PyCharm console.

Run the Task Management System from PyCharm

1. Run the application by clicking the play ▶ button (Run 'Flask (app.py)' Shift+F10 on Linux and Windows, ^R on macOS).

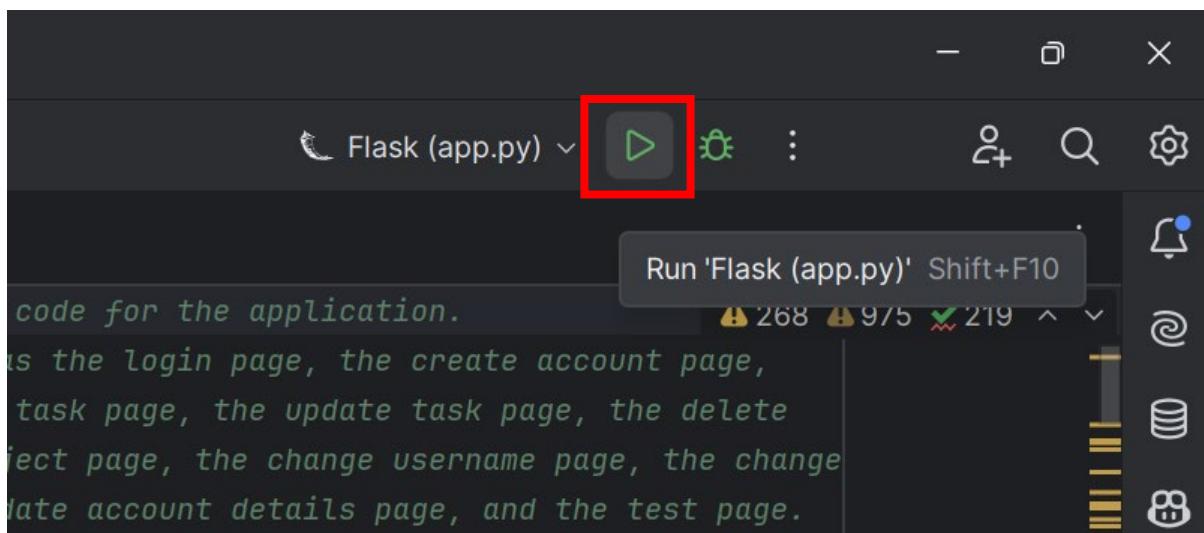


Figure 66 - Click the Run button (resembles a green play button).

2. Click the blue link (<http://<ip-address>>) in the Python console.

```
Run  Flask (app.py) ×
C:\Users\rober\Newfolder6\task-management-system-clean-install\venv\Scripts\python.exe -m flask run
Could not find platform independent libraries <prefix>
* Serving Flask app 'app.py'
* Debug mode: on
WARNING: This development server is not secure.
Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
Could not find platform independent libraries <prefix>
* Debugger is active!
* Debugger PIN: 556-336-442
task-management-system-clean-install > 🐍 app.py
1:1
```

Figure 67 - Task Management system is available at the blue link.

3. The Task Management System is now available in your web browser.

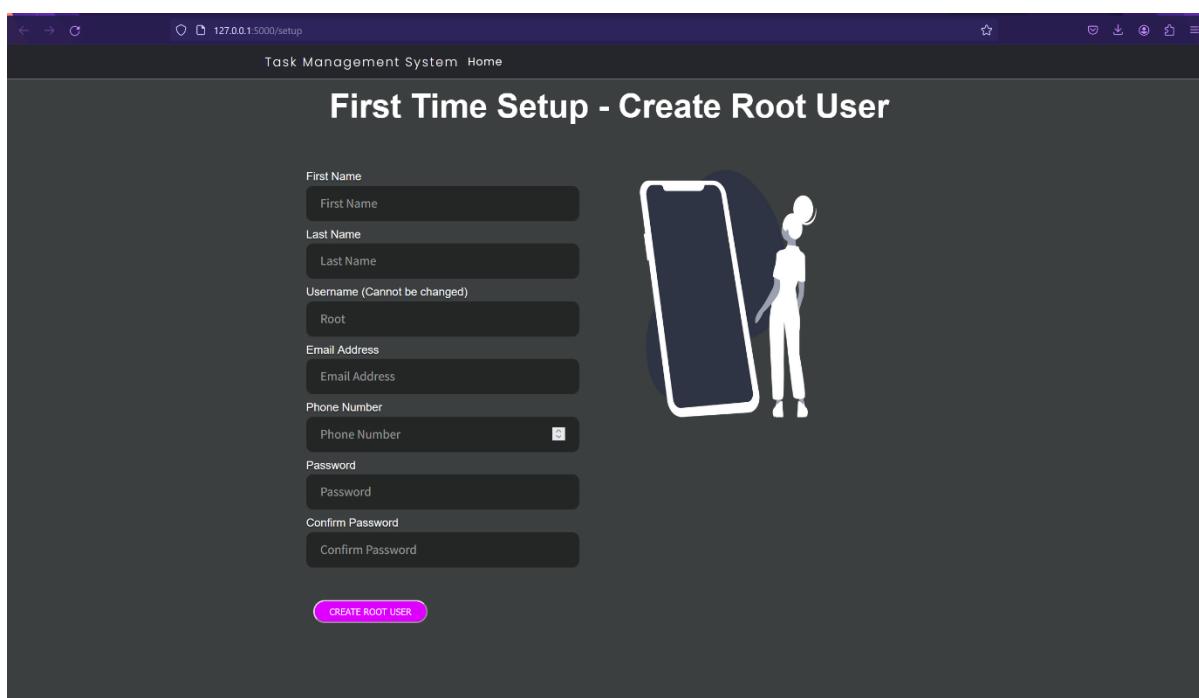


Figure 67 - Task Management System Setup page.

4. Troubleshooting tip: If you get an “Address already in use” error, this means the port is already in use by another application or process.

```
Run  app ×
C:\Users\alexanderrobertson\Downloads\newFolder3\task-management-system-clean-install\venv\bin\python /Users\alexanderrobertson\Downloads\newFolder3\task-management-system-clean-install\app.py
* Serving Flask app 'app'
* Debug mode: on
Address already in use
Port 5000 is in use by another program. Either identify and stop that program, or start the server with a different port.
On macOS, try disabling the 'AirPlay Receiver' service from System Preferences -> General -> AirDrop & Handoff.
Process finished with exit code 1
task-management-system-clean-install > 🐍 app.py
1645:41  LF  UTF-8  4 spaces  Python 3.12 (task-manage...system-clean-install) (3)
```

Figure 68 - Error if the port is already in use.

To fix this, change the port number in the app.py code below the `if __name__ == '__main__':` function, Line 1645 (may be different if you have modified the code) `port =`

```
int(os.environ.get('PORT', 5000)) #port number for example to port =
int(os.environ.get('PORT', 5100)) #port number.
```

```
1641
1642
1643 > if __name__ == '__main__':
1644     '''This is the main function that runs the application.'''
1645     port = int(os.environ.get('PORT', 5000)) #port number
1646     app.run(debug=True, host='0.0.0.0', port=port) #run the application
1647
1648
1649
```

Figure 69 - Old port number 5000

```
1642
1643 > if __name__ == '__main__':
1644     '''This is the main function that runs the application.'''
1645     port = int(os.environ.get('PORT', 5100)) #port number
1646     app.run(debug=True, host='0.0.0.0', port=port) #run the application
1647
1648
1649
```

Figure 70 - New port number 5100.

Setting up and pushing to Git repository (Version Control)

1. Make sure hidden files are visible on your computer's file browser.
 - a. Windows:
 - i. Select the ... icon in File Explorer.

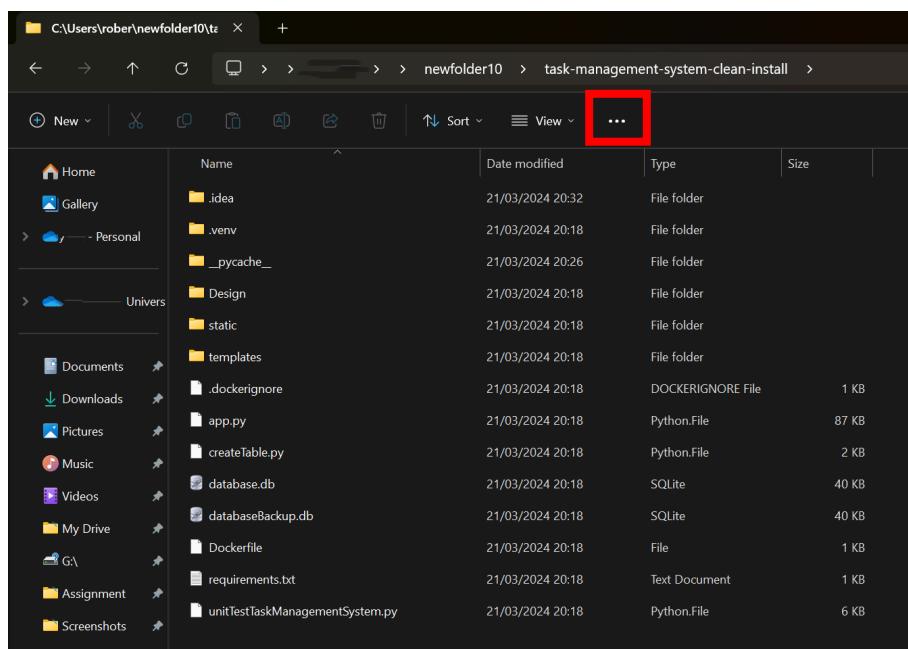


Figure 71 - ... icon for menu

- ii. Select “⚙️ Options” from the drop-down menu.

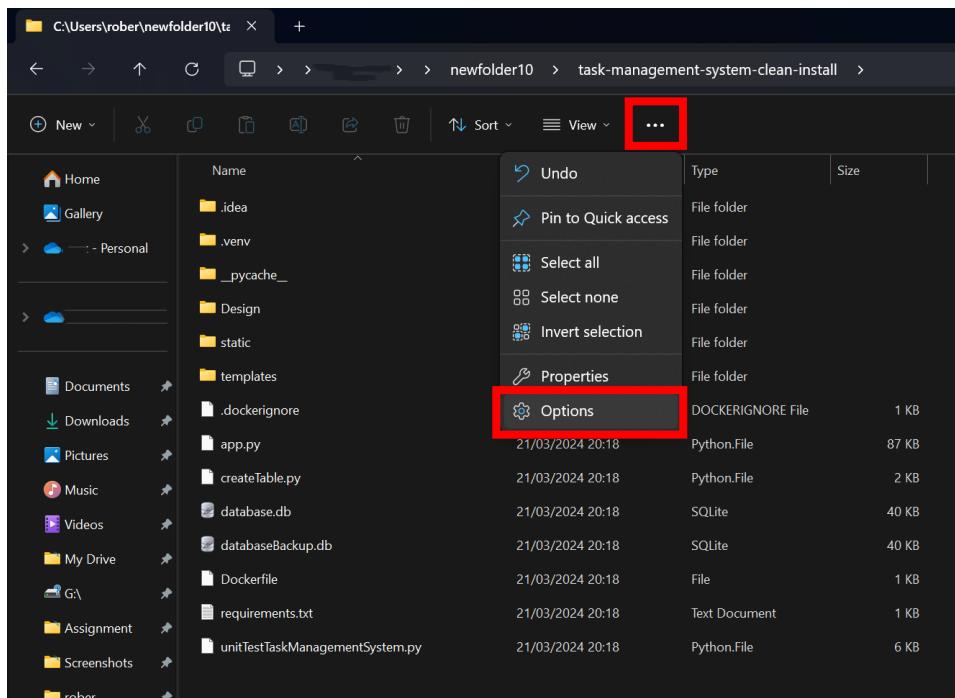


Figure 72 - File Explorer dropdown menu.

- iii. Select "View" then "Show hidden files, folders and drives" radio button and then click "OK".

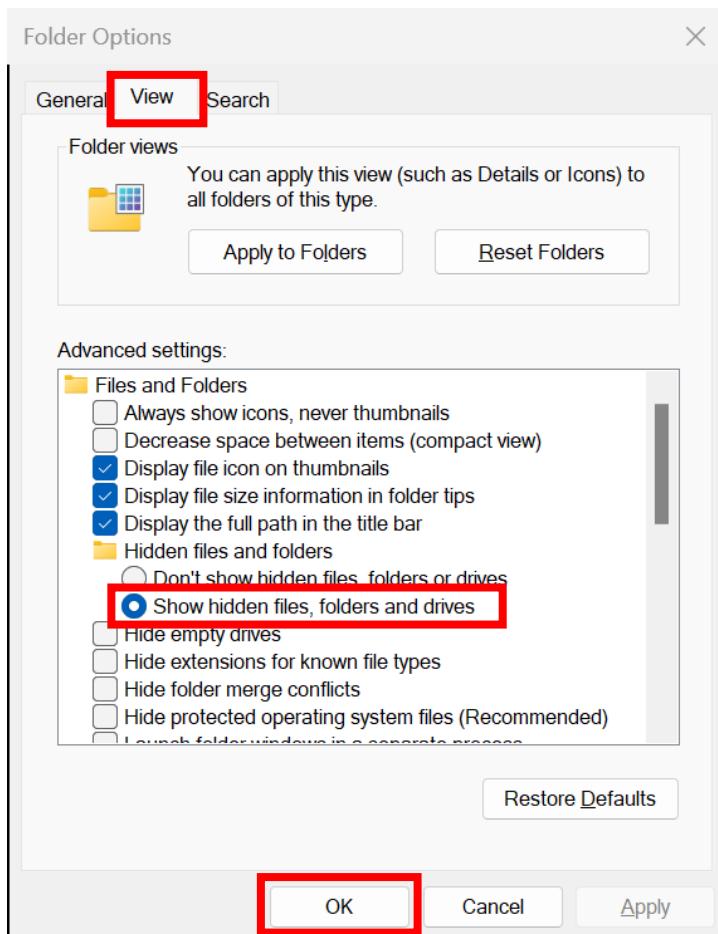


Figure 73 - Folder Options - Select "Show hidden files, folders and drives".

iv. The .git folder will now be visible.

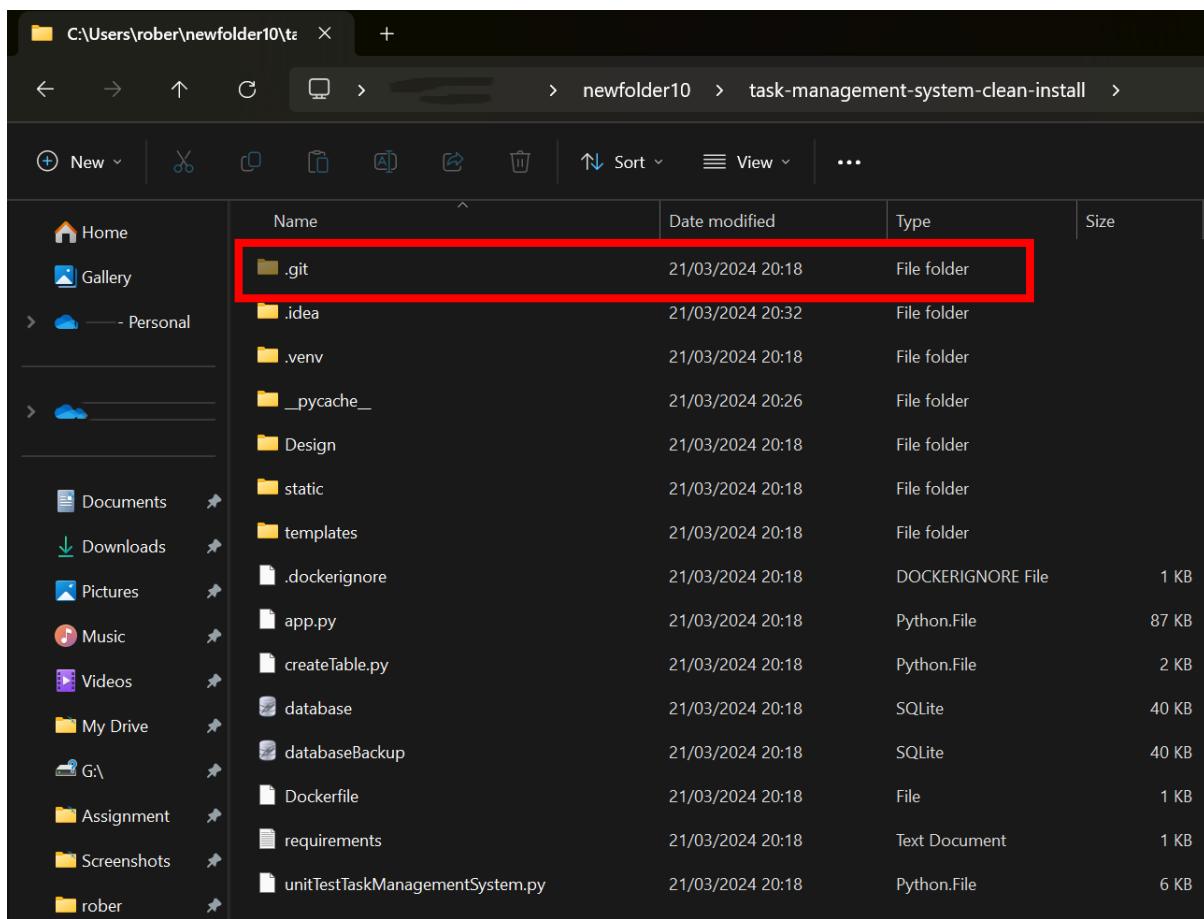


Figure 74 - .git folder in task-management-system (clean-install) directory.

- b. macOS:
 - i. Navigate to the task-management-system or task-management-system-clean-install directory in Finder.
 - ii. Press 'Command' + 'Shift' + . (Full Stop key).
 - iii. The .git folder will be visible.
 - c. Linux:
 - i. Enter the command "ls -a" into the Terminal in the task-management-system or task-management-system-clean-install directory and .git will be visible.
 - ii. Press 'Ctrl+H' in File Manager to view hidden files and folders that will make the .git folder visible.
2. Delete the existing .git folder.

S275931

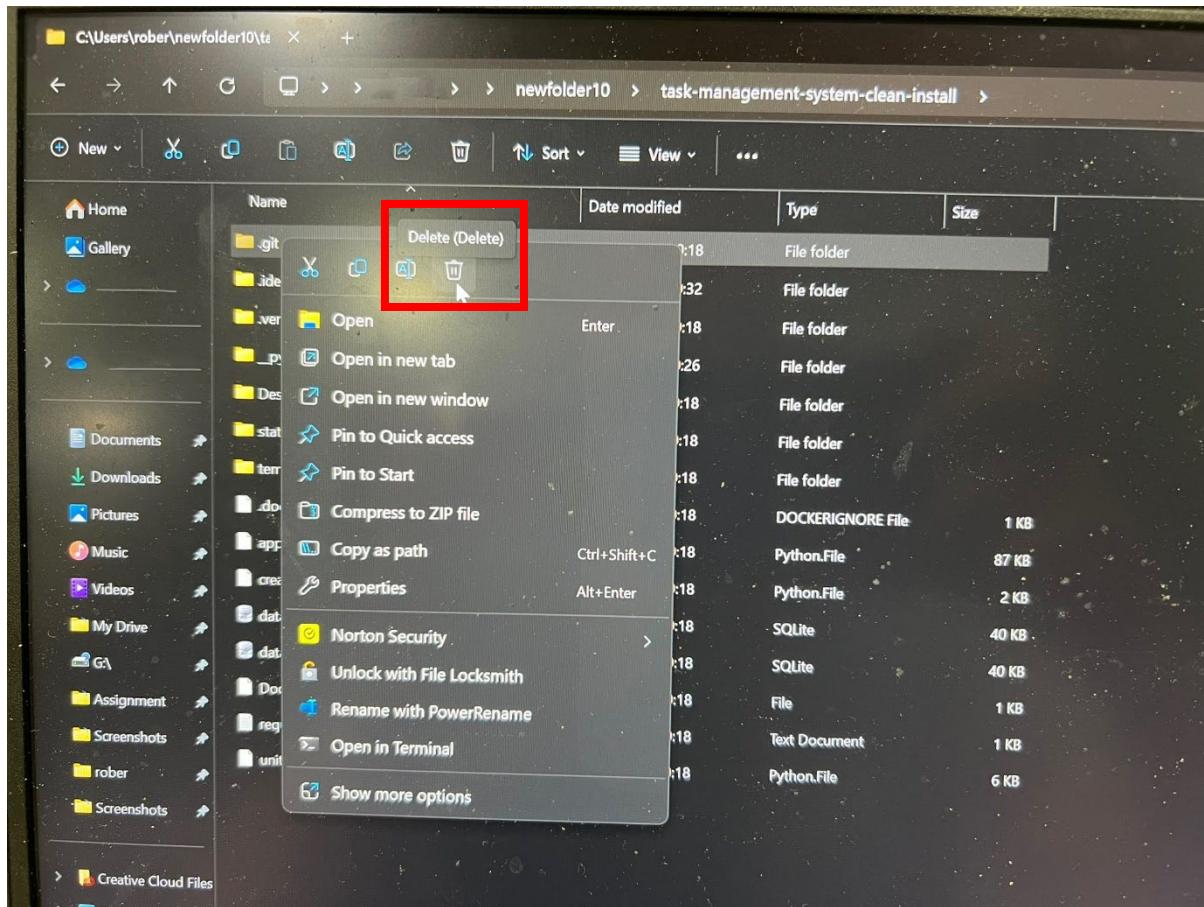


Figure 75 - Deleting the .git folder.

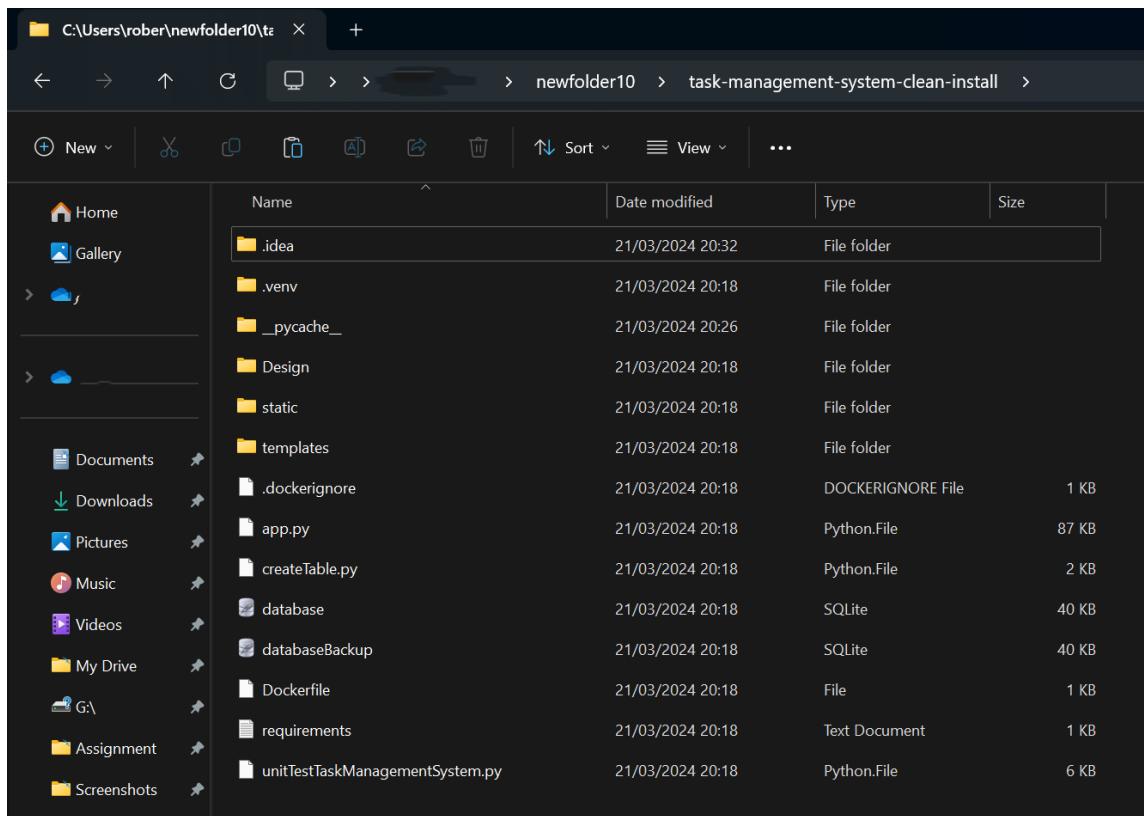


Figure 76 - .git folder deleted.

3. Create a GitHub account if you haven't already got one. You can also choose to host your Git repository elsewhere such as on GitLab, Amazon AWS CodeCommit, Bitbucket, Microsoft Azure DevOps, SourceForge etc. GitHub will be used for demonstration purposes.
4. Create a git repository on GitHub.com (or other Git version control platform)
 - a. On your GitHub Dashboard, click "New" next to "Top Repositories".

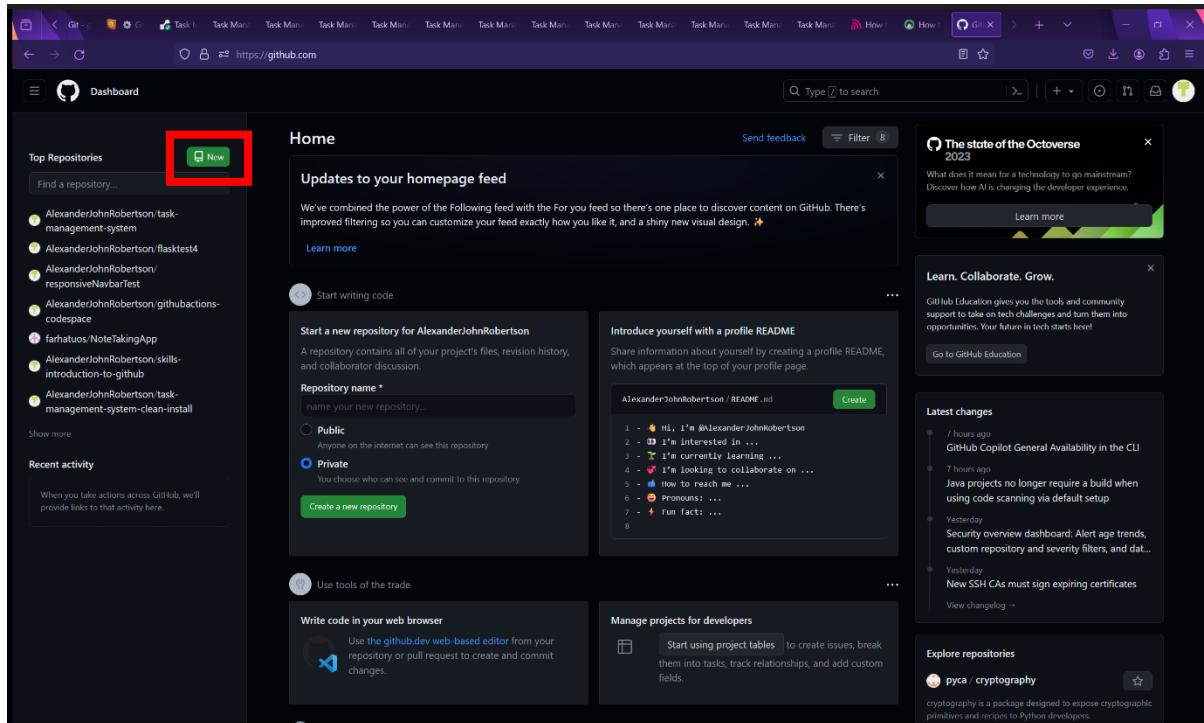


Figure 77 - The GitHub dashboard

- b. Choose a name for your GitHub repository, add a description (optional), choose if you want to make your repository public or private, choose license (optional) and then click "Create Repository".

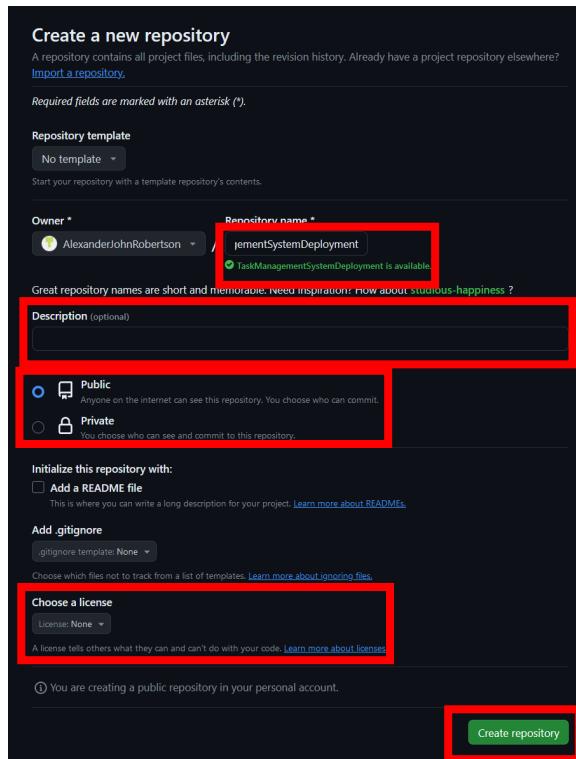


Figure 78 - Creating a new repository.

c. Your GitHub repository will now be created.

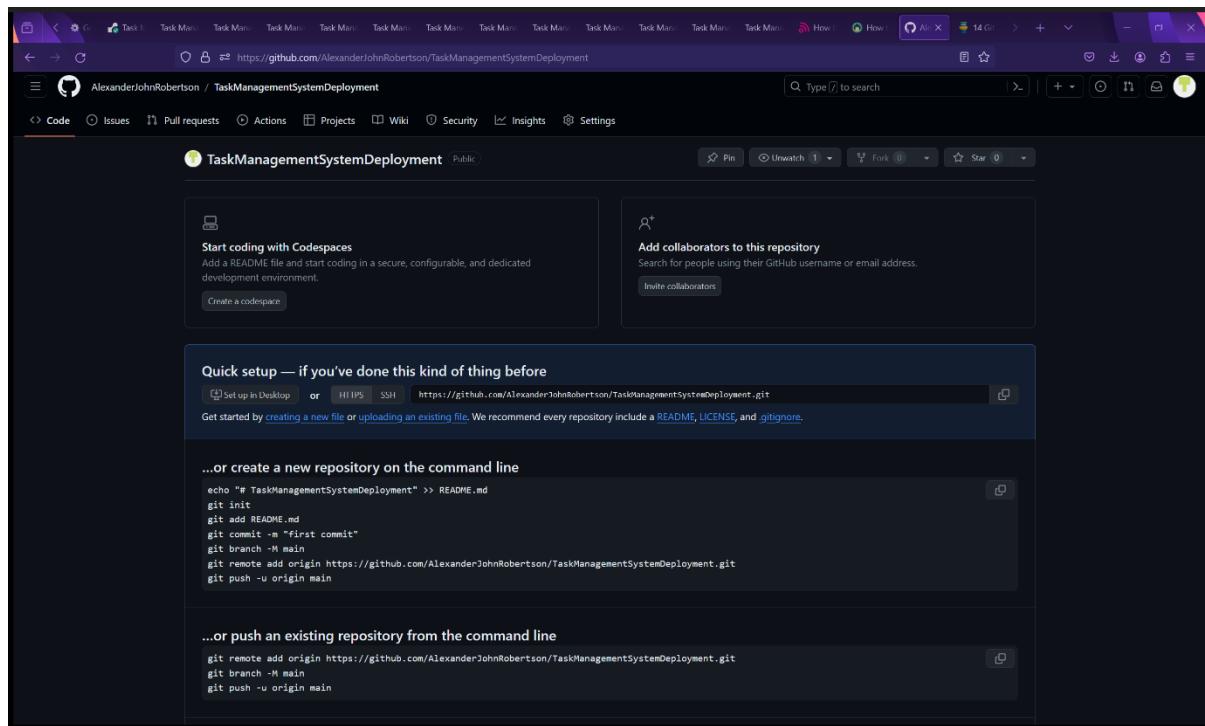
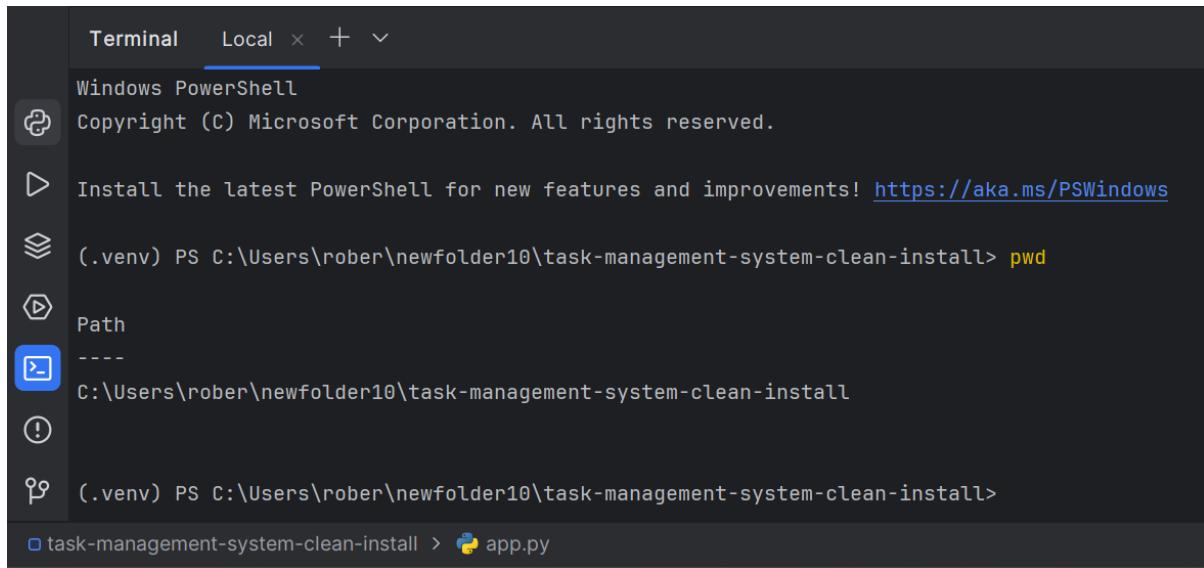


Figure 79 - New GitHub repository with Git command instructions.

5. Make sure the current working directory matches the location of the Task Management System source code you have downloaded by entering the command “pwd” into the PyCharm terminal.



The screenshot shows a PyCharm terminal window titled "Terminal". The tab bar has "Local" selected. The terminal output shows:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> pwd
Path
-----
C:\Users\rober\newfolder10\task-management-system-clean-install

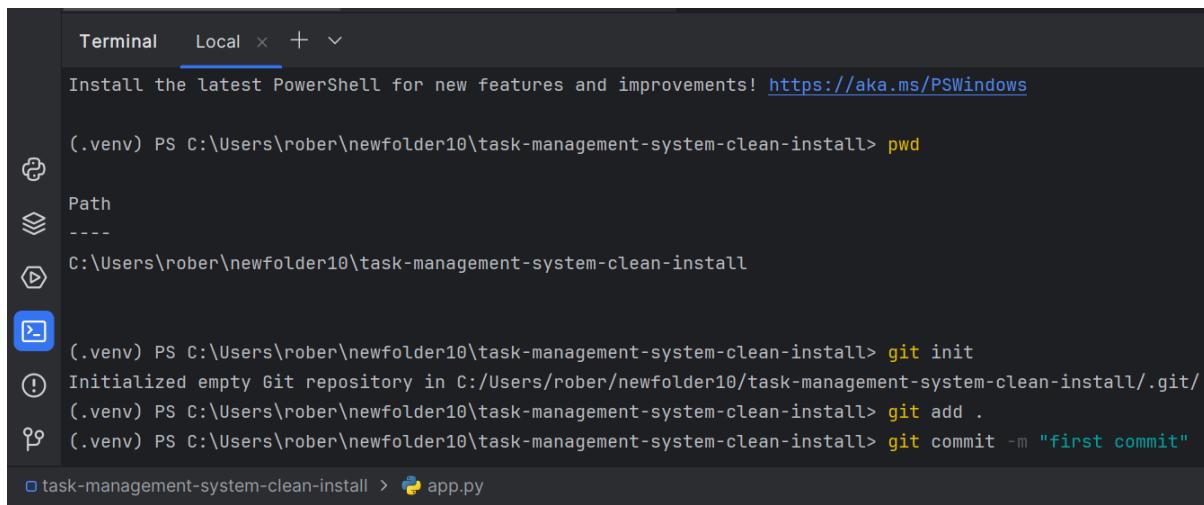
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install>

```

The status bar at the bottom shows "task-management-system-clean-install > 🐍 app.py".

Figure 80 - Making sure you are in the correct working directory.

6. Set up the local Git repository in PyCharm by entering the following Git commands into the PyCharm terminal in the following order:
 - a. `git init`
 - b. `git add .`
 - c. `git commit -m "first commit"`
 - d. `git branch -M main`
 - e. `git remote add origin https://github.com/<GitHub-username>/<git-repository-name>.git`
 - f. `git push -u origin main`



The screenshot shows a PyCharm terminal window titled "Terminal". The tab bar has "Local" selected. The terminal output shows:

```

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> pwd
Path
-----
C:\Users\rober\newfolder10\task-management-system-clean-install

(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git init
Initialized empty Git repository in C:/Users/rober/newfolder10/task-management-system-clean-install/.git/
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git add .
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git commit -m "first commit"


```

The status bar at the bottom shows "task-management-system-clean-install > 🐍 app.py".

Figure 81 - Setting up the Git repository in PyCharm console.

```

Terminal Local × + ↻
create mode 100644 templates/viewusers.html
create mode 100644 unittestTaskManagementSystem.py
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git branch -M main
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git remote add origin https://github.com/AlexanderJohnRobertson/TaskManagementSystemDeployment.git
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install> git push -u origin main
Enumerating objects: 1582, done.
Counting objects: 100% (1582/1582), done.
Delta compression using up to 32 threads
Compressing objects: 100% (1582/1582), done.
Writing objects: 100% (1582/1582), 23.34 MiB | 19.65 MiB/s, done.
Total 1582 (delta 104), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (104/104), done.
To https://github.com/AlexanderJohnRobertson/TaskManagementSystemDeployment.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
(.venv) PS C:\Users\rober\newfolder10\task-management-system-clean-install>

```

task-management-system-clean-install > app.py

28:1 28 CRLF UTF-8 4 spaces Python 3.12 (task)

Figure 82 - Setting up the Git repository in PyCharm console.

7. Refresh the GitHub repository page on [github.com](https://github.com/AlexanderJohnRobertson/TaskManagementSystemDeployment) to view the contents that you have pushed to your repository.

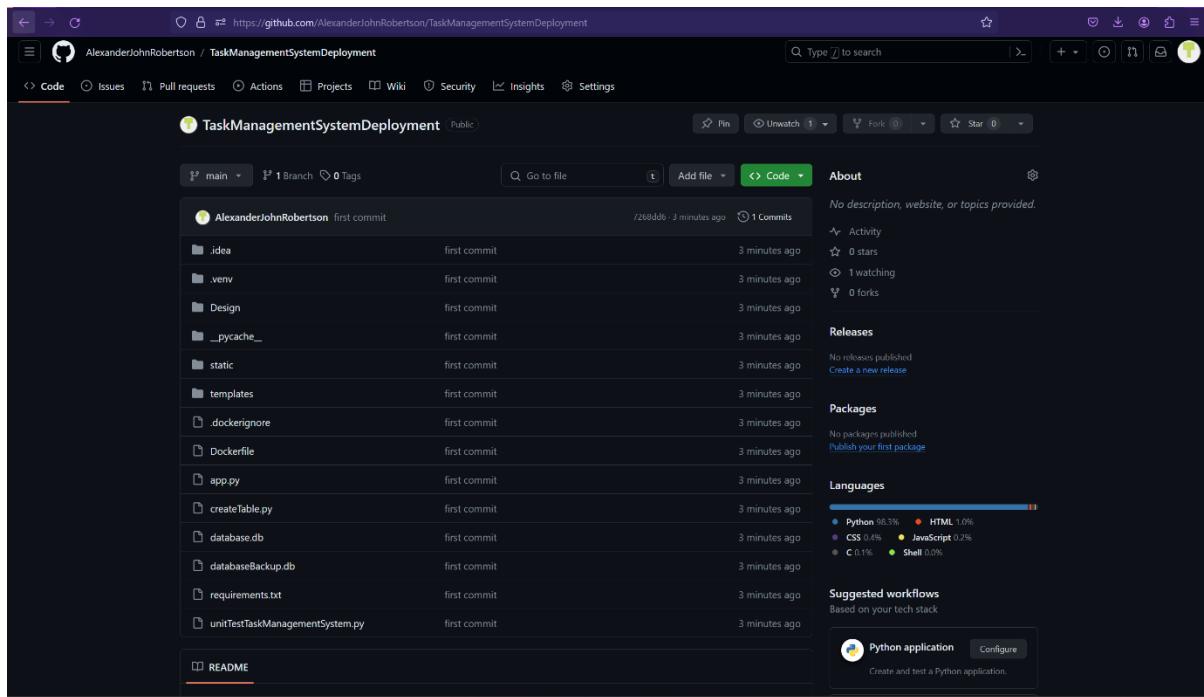


Figure 83 - Newly created Git Repository with source code and program files.

8. Your Git repository is now ready to use.
9. For future commits to your Git repository, you only need to enter the three following Git commands into the PyCharm terminal each time:
 - git add .
 - git commit -m "<your message here>"
 - git push origin main

Modifying Source code, Adding and Testing New Features

WARNING: ONLY MODIFY THE SOURCE CODE IF YOU HAVE EXPERIENCE WITH AND UNDERSTAND PYTHON 3, SQL, SQLITE, HTML, CSS, JAVASCRIPT, FLASK, DOCKER AND DOCKERFILES, GUNICORN AND GIT. EDITING THESE FILES WITHOUT THE RIGHT KNOWHOW MAY DAMAGE THE TASK MANAGEMENT SYSTEM, RENDERING THE APPLICATION INOPERABLE AND/OR CORRUPT THE

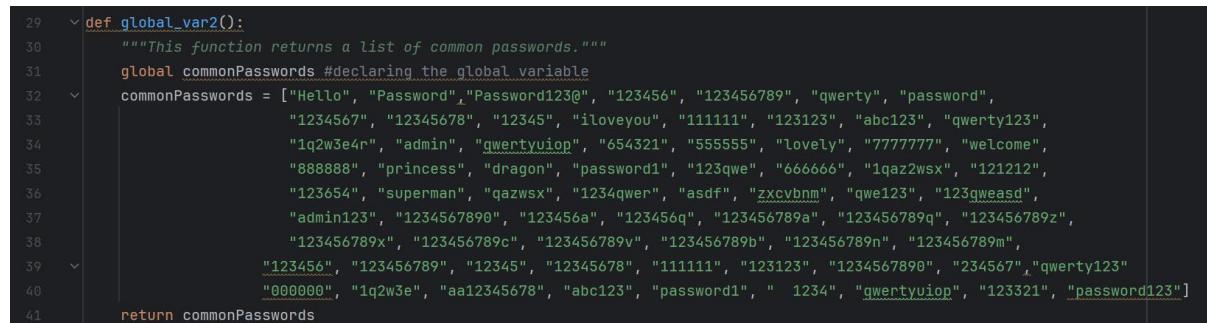
DATABASE RESULTING IN PERMANENT DATA LOSS! ALWAYS BACK UP THE ORIGINAL APPLICATION AND USE VERSION CONTROL ON YOUR OWN GIT REPOSITORY IN CASE THE APPLICATION GETS DAMAGED. THE SOFTWARE AUTHOR CANNOT BE HELD RESPONSIBLE FOR DAMAGE CAUSED BY A THIRD PARTY MODIFYING THE SOURCE CODE.

Suggested Modifications

Editing the List of Common Password to be Banned

You can change the list of common passwords in the `commonPasswords` global variable.

Open app.py in PyCharm edit the global variable on lines 29-41, save the edit, run the application to test it, if it works, you can commit to your git repository or Docker repository for deployment.



```

29  def global_var2():
30      """This function returns a list of common passwords."""
31      global commonPasswords #declaring the global variable
32      commonPasswords = ["Hello", "Password", "Password123@", "123456", "123456789", "qwerty", "password",
33                           "1234567", "12345678", "12345", "iloveyou", "111111", "123123", "abc123", "qwerty123",
34                           "1q2w3e4r", "admin", "qwertyuiop", "654321", "555555", "lovely", "7777777", "welcome",
35                           "888888", "princess", "dragon", "password1", "123qwe", "666666", "1qaz2wsx", "121212",
36                           "123654", "superman", "qazwsx", "1234qwer", "asdf", "zxcvbnm", "qwe123", "123qweasd",
37                           "admin123", "1234567890", "123456a", "123456q", "123456789a", "123456789q", "123456789z",
38                           "123456789x", "123456789c", "123456789v", "123456789b", "123456789n", "123456789m",
39                           "123456", "123456789", "12345", "12345678", "111111", "123123", "1234567890", "234567", "qwerty123"
40                           "000000", "1q2w3e", "aa12345678", "abc123", "password1", "1234", "qwertyuiop", "123321", "password123"]
41      return commonPasswords

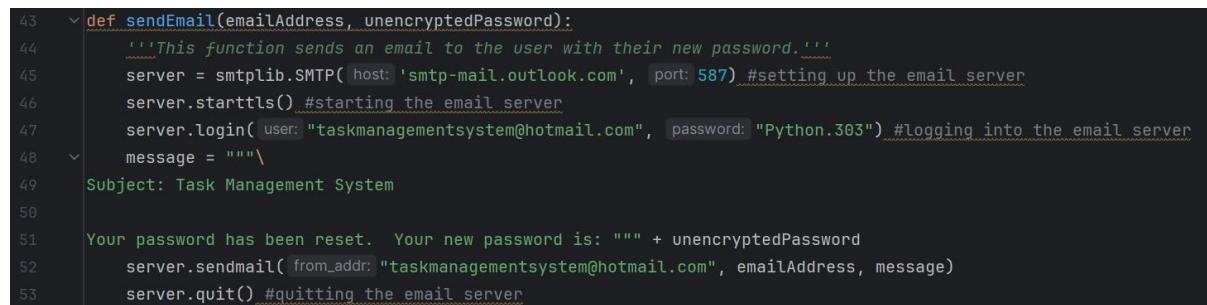
```

Figure 84 - List of banned common passwords.

Changing the Email Content

You can change the email domain, email address, email password and email content for the automated emails that are sent during setup and when an administrator resets a user password.

Open app.py in PyCharm edit the `sendEmail` function on lines 43-53 for the email that is sent when an administrator resets a user's password and the `setupEmail` function on lines 55-65. You can change save the edit, run the application to test it, if it works, you can commit to your git repository or Docker repository for deployment.



```

43  def sendEmail(emailAddress, unencryptedPassword):
44      """This function sends an email to the user with their new password."""
45      server = smtplib.SMTP( host: 'smtp-mail.outlook.com', port: 587) #setting up the email server
46      server.starttls() #starting the email server
47      server.login( user: "taskmanagementsystem@hotmail.com", password: "Python.303") #logging into the email server
48      message = """
49      Subject: Task Management System
50
51      Your password has been reset. Your new password is: """ + unencryptedPassword
52      server.sendmail( from_addr: "taskmanagementsystem@hotmail.com", emailAddress, message)
53      server.quit() #quitting the email server

```

Figure 85 - Email that is automatically sent to the user when an administrator resets another user's password.

```

55  def setupEmail(emailAddress, unencryptedPassword):
56      '''This function sends an email to the root user with their new password.''''
57      server = smtplib.SMTP(host='smtp-mail.outlook.com', port=587) #setting up the email server
58      server.starttls() #starting the email server
59      server.login(user="taskmanagementsystem@hotmail.com", password="Python.303") #logging into the email server
60      message = """
61      Subject: Task Management System
62
63      Welcome to the Task Management System (first time setup). Your password is: """ + unencryptedPassword
64      server.sendmail(from_addr="taskmanagementsystem@hotmail.com", emailAddress, message)
65      server.quit() #quitting the email server

```

Figure 86 - Email that is automatically sent to the Root User when they set up the Task Management System.

Testing Experimental Features

You can test new features in the Task Management System's source code. Add functionality (Backend code) to experimental features in the three test page functions (testpage (line 1626), testpage2 (line 1631) and testpage3 (line 1637)) in app.py. Every function has a Flask route() decorator above it to specify the paths for the test page URLs.

```

1625     @app.route(rule='/testpage', methods=['GET', 'POST'])
1626     def testpage():
1627         '''This is a test page for testing purposes.'''
1628         return render_template('testpage.html')

```

Figure 87 - Flask route() decorators with test page functions. Python and SQLite code for the test pages goes in these functions.

```

1630     @app.route('/testpage2')
1631     def testpage2():
1632         '''This is a test page for testing purposes.'''
1633         for i in range(10):
1634             print("Hello World")
1635         return render_template('testpage2.html')

```

Figure 88 - Flask route() decorators with test page functions. Python and SQLite code for the test pages goes in these functions.

```

1637     @app.route('/test3')
1638     def testpage3():
1639         '''This is a test page for testing purposes.'''
1640         return render_template('test3.html')

```

Figure 89 - Flask route() decorators with test page functions. Python and SQLite code for the test pages goes in these functions.

The frontend HTML files for the webpages and forms are in the templates folder.

You can test new features on testpage.html, testpage2.html and test3.html. You can also test new user interface designs here. If you wish to test new changes to the CSS files for the UI, you should create a test CSS file, link it to the test page and test CSS style changes to the UI in the test copies of the CSS files to prevent unwanted changes and damage to the main files.

```
1 <!-- test page to experiment with and test new features -->
2 <head>
3   <link rel="stylesheet" href="/static/createaccountloginformstyle.css" >
4   <link rel="stylesheet" href="/static/formstyle.css" >
5 </head>
6 {% extends 'base3.html' %}
7
8 {% block content %}
9   <h2>This is a test page to experiment with and test new features</h2>
10 <h2>Create Account</h2>
11 <form method="post">
12   <section1 class="section1 left">
13     <div class="input-container">
14       <label class="text1">First Name</label>
15       <input type="text" id="firstname" name="firstname" placeholder="First Name" />
16     </div>
17     <div class="input-container">
18       <label for="lastname." required>Last Name</label>
19       <input type="text" id="lastname" name="lastname" placeholder="Last Name" />
20     </div>
21     <div class="input-container">
22       <label for="username.">Username</label>
23       <input type="text" id="username" name="username" placeholder="Username" />
24     </div>
25     <div class="input-container">
26       <label for="email.">Email Address</label>
27       <input type="email" id="email" name="email" placeholder="Email Address" />
28     </div>
29     <div class="input-container">
```

Figure 90 - HTML code on test page.

Once satisfied with feature tests and UI tests, you can integrate the changes into the main application features.

Unit Testing the Task Management System Source Code

1. Open `unitTestTaskManagementSystem.py` in PyCharm.

The screenshot shows a Microsoft Visual Studio Code interface with the following details:

- Project Explorer:** Shows files like `changeuserpassword.html`, `deleteaccount.html`, `changeaccounttype.html`, `changeuserpassword.html`, and `unitTestTaskManagementSystem.py`.
- Code Editor:** Displays the `unitTestTaskManagementSystem.py` file containing Python test code for a Task Management System. The code uses the `unittest` module to test routes for login, account creation, and task addition.
- Terminal:** Shows a Windows PowerShell session with the command `Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows`.
- Status Bar:** Shows the path `(.venv) PS C:\Users\rcober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2` and system information like CPU, memory usage, and Python version.

Figure 91 - unitTestTaskManagementSystem.py open in PyCharm for unit testing the Task Management System.

2. Change “TaskManagementSystem” or “TaskManagementSystem_Clean_Install” to “Current File”.

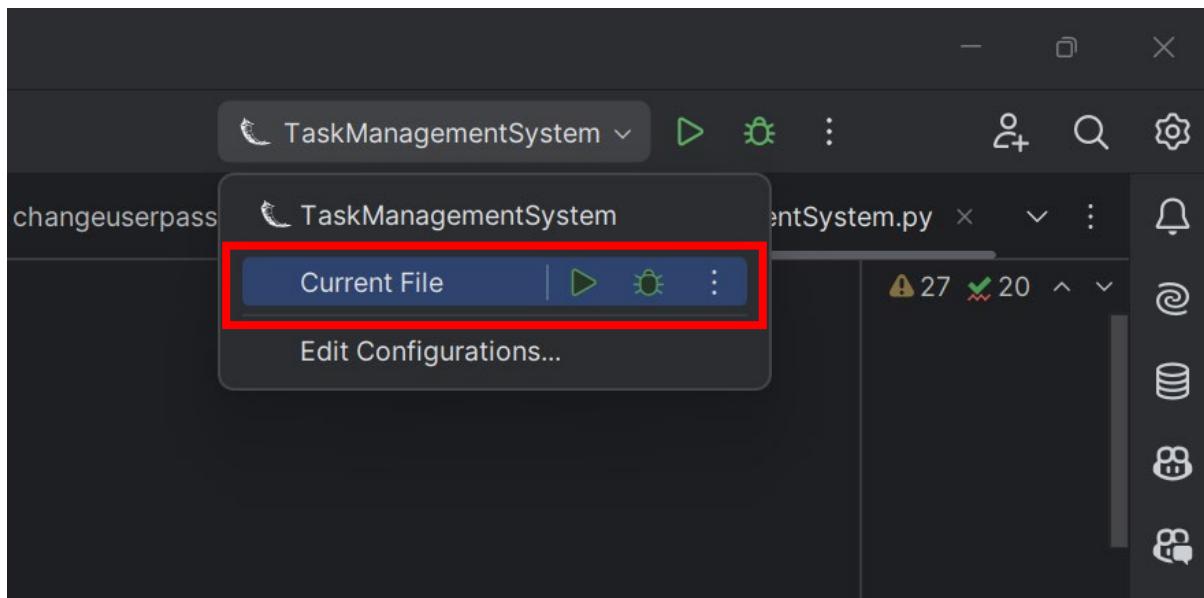


Figure 92 - Select "Current File" instead of "Task Management System".

3. Run the current file.

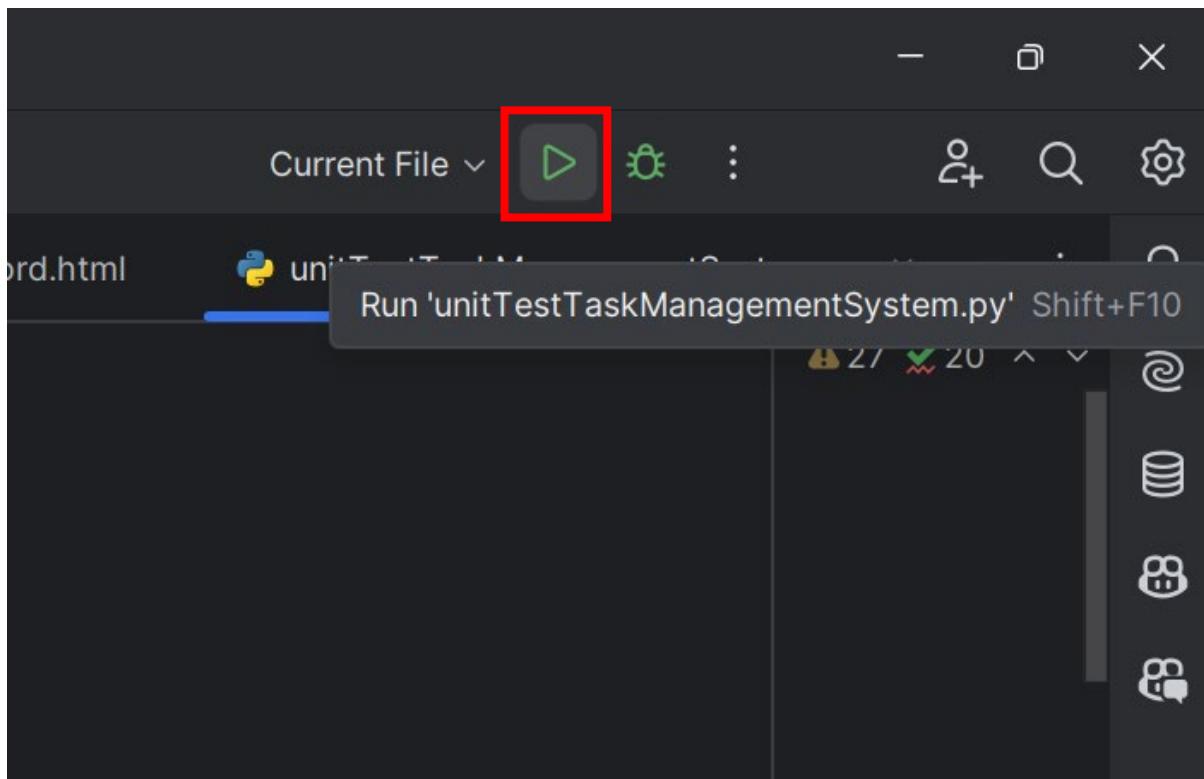


Figure 93 - Click the Run button.

4. If the Taks Management System passes all the unit tests, then you will receive a message saying “OK” in the Python console.

The screenshot shows a terminal window titled "Run" with the command "Python tests in unittestTaskManagementSystem.py". The output indicates "Tests passed: 22 of 22 tests - 30 ms". Below this, it says "Launching unittests with arguments python -m unittest C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2\unitTestTaskManagementSystem.py". The terminal then displays "Ran 22 tests in 0.053s", "OK", and "Process finished with exit code 0". The status bar at the bottom shows "91:45" and "Python 3.12 (TaskManagementSystem2)".

Figure 94 - Passed all elements of the Unit Testing.

5. If the Task Management System fails one or more of the unit tests, you will receive an error message in the Python console saying “Failure”, “<Error Message>”, “FAILED”. A Test Results tree will appear next to the Python console that navigates to the unit test that failed.

The screenshot shows a terminal window titled "Run" with the command "Python tests in unittestTaskManagementSystem.py". The output indicates "Tests failed: 1, passed: 21 of 22 tests - 34 ms". Below this, it says "Launching unittests with arguments python -m unittest C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2\unitTestTaskManagementSystem.py". The terminal then displays "Failure", followed by a detailed traceback: "Traceback (most recent call last): File "C:\Users\rober\OneDrive\Documents\Advanced Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2\unitTestTaskManagementSystem.py", line 10, in <module> self.assertIn(b'Access Hello\$\$\$\$ Denied', response.data)", and "AssertionError: b'Access Hello\$\$\$\$ Denied' not found in b'<!DOCTYPE html><html lang="en"><head>'. The terminal also shows "Ran 22 tests in 0.043s", "FAILED (failures=1)", and "Process finished with exit code 1". The status bar at the bottom shows "58:41" and "Python 3.12 (TaskManagementSystem2)".

Figure 95 Failed the Unit Testing as the Task Management System has failed one or more unit test elements.

6. Fix the code that is causing the unit test to fail.
7. Re-Run the unit tests.
8. Add new unit tests to the TestRoutes class below the setup function to unit test new features.

```

  ↳ AlexanderJohnRobertson +1
6 ▶ v class TestRoutes(unittest.TestCase):
    ↳ AlexanderJohnRobertson +1
7 ⚡↑ v     def setUp(self): # Set up the app for testing
8         self.app = app.test_client()
9
10 ↳ AlexanderJohnRobertson +1
10 ▶ v     def test_login(self): # Test the login route
11         response = self.app.get('login')
12         self.assertEqual(response.status_code, second: 200)
13         self.assertIn( member: b'Log', response.data)
14
15 ↳ AlexanderJohnRobertson +1
15 ▶ v     def test_create_account(self): # Test the create account route
16         response = self.app.get('createaccount')
17         self.assertEqual(response.status_code, second: 200)
18         self.assertIn( member: b'Create', response.data)
19

```

Figure 96 - Snippet of the Unit Testing source code.

Deploy online using Render.com or other webhosting service using a Continuous Integration / Continuous Deployment (CI/CD) Pipeline.

This section is for deploying the Task Management System online using a webhosting service using a Continuous Integration / Continuous Deployment (CI/CD) Pipeline. Render.com will be used in the given instructions however other platforms could also be used.

1. Go to Render.com: <https://render.com/>
2. Create a Reender.com account if you don't already have one or sign in to your existing account.
3. Go to your Render.com Dashboard.

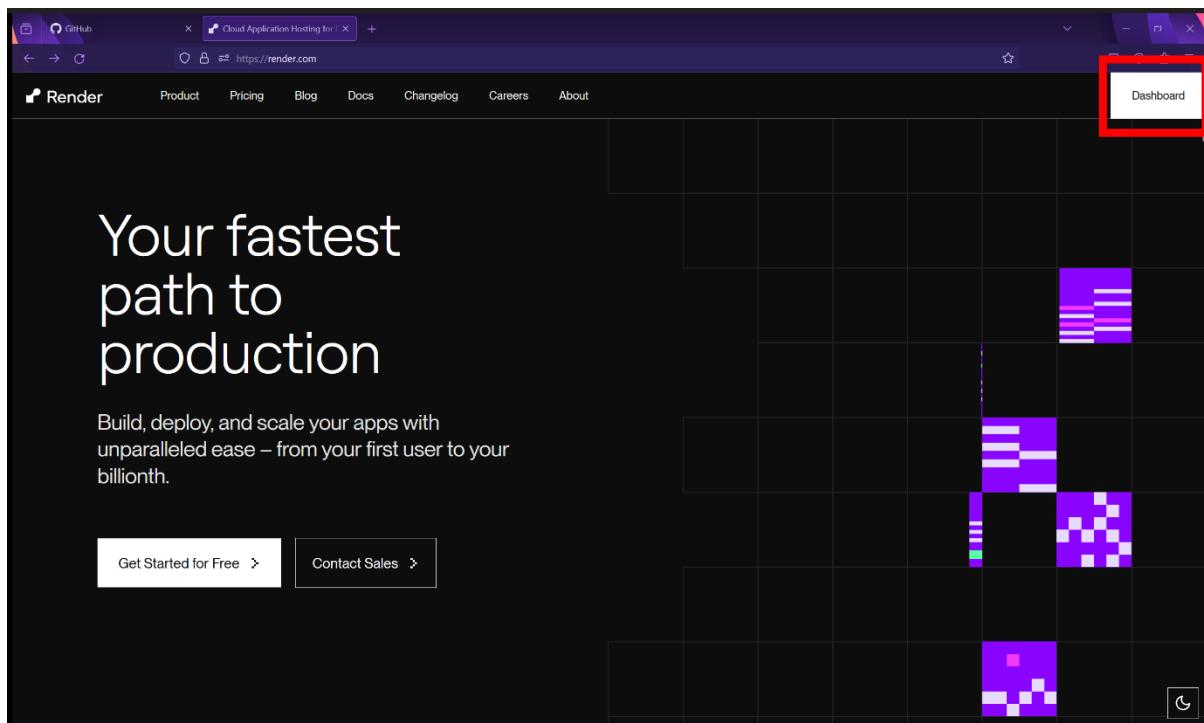


Figure 97 - Render.com homepage. Select “Dashboard”.

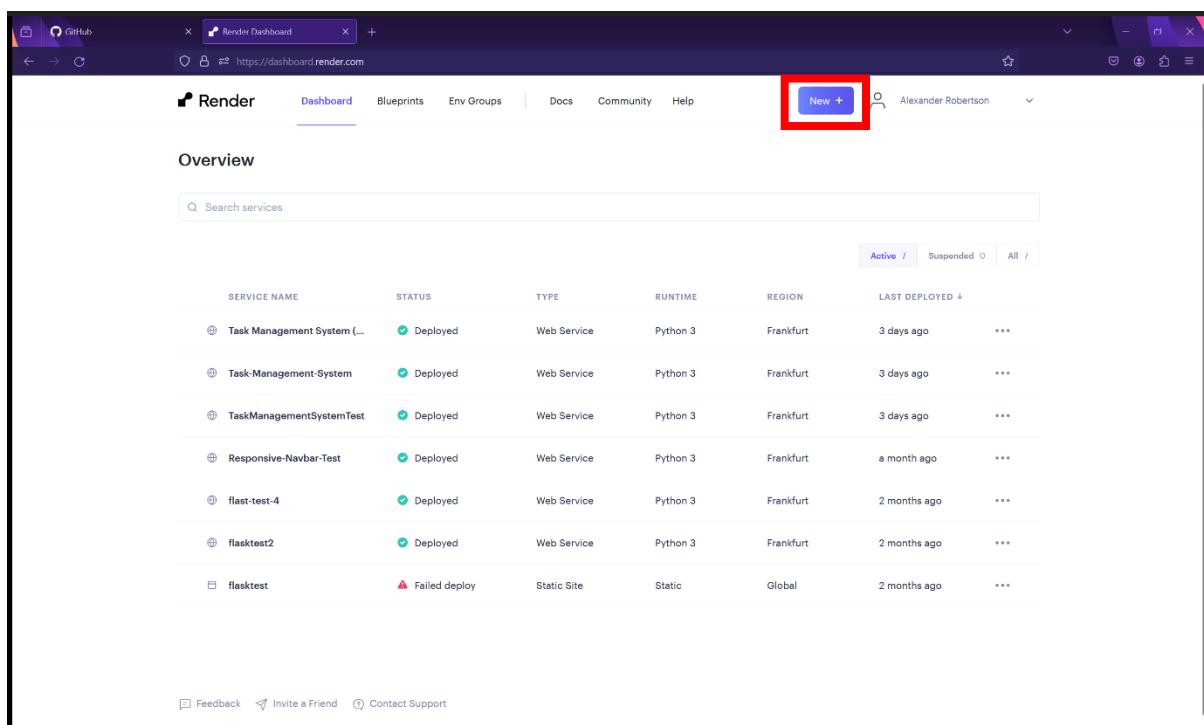


Figure 98 - Render.com Dashboard

4. Select “New” then “Web Service”.

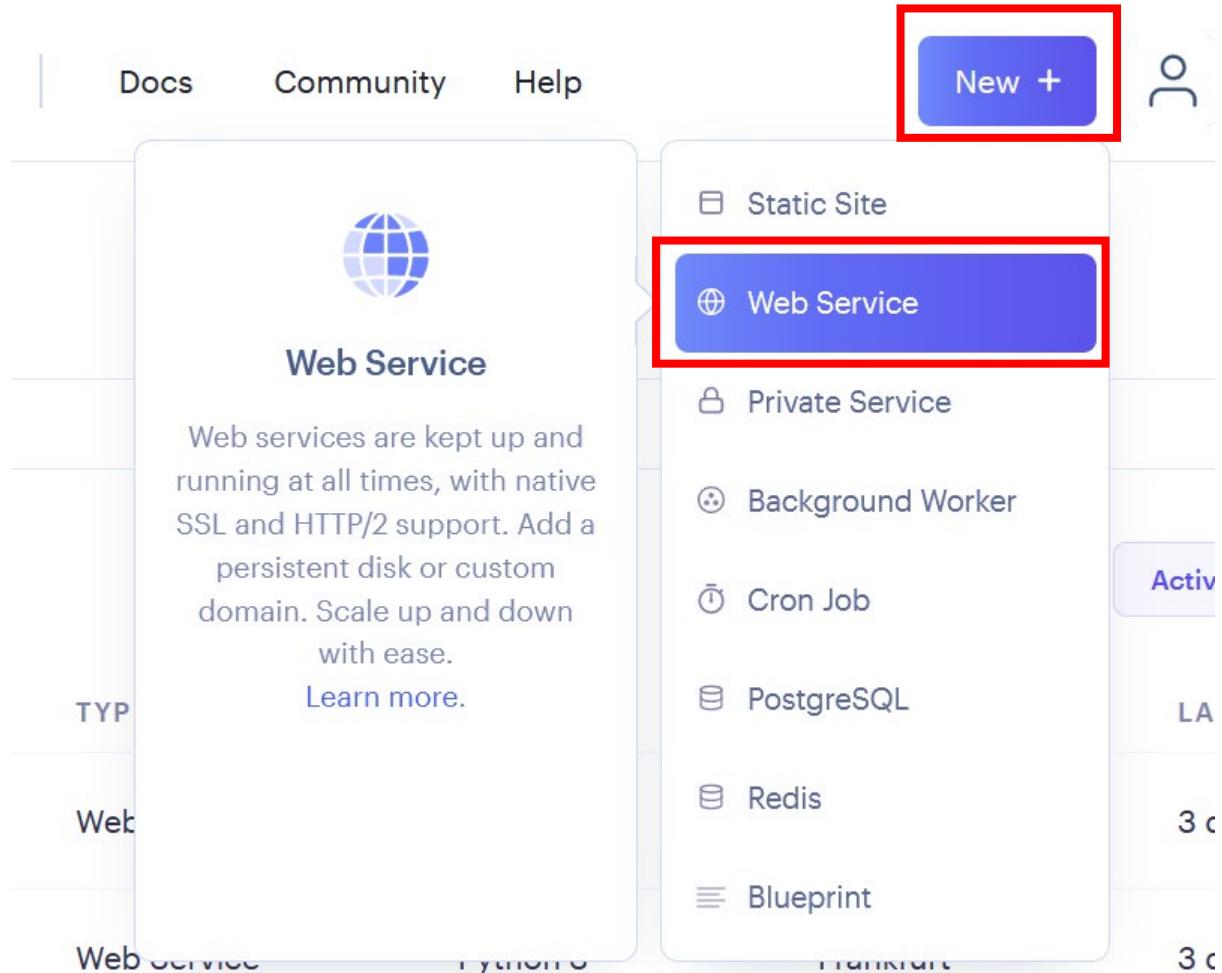


Figure 99 - Select "New Service" from the "New+" dropdown menu.

5. Select "Build and deploy from a Git repository" and then click "Next".

The screenshot shows the 'Create a new Web Service' form. At the top, there's a header 'Create a new Web Service' and a note: 'Connect a Git repository, or use an existing image.' Below this, a question 'How would you like to deploy your web service?' is followed by two options:

- Build and deploy from a Git repository
Connect a GitHub or GitLab repository.
- Deploy an existing image from a registry ADVANCED
Pull a public image from any registry or a private image from Docker Hub, GitHub, or GitLab.

A large red box surrounds the first option. At the bottom right, a blue 'Next' button is highlighted with a red box.

Figure 100 - Select "Build and deploy from a Git repository" then click "Next".

6. Connect the Git repository that you created in the previous steps for Setting up and pushing to a Git repository. Choose the repository from the list then click “Connect”.

Connect a repository

Search...

- AlexanderJohnRobertson / TaskManagementSystemDeployment · 3 days ago Connect
- AlexanderJohnRobertson / task-management-system-clean-install · 3 days ago Connect
- AlexanderJohnRobertson / task-management-system · 3 days ago Connect
- AlexanderJohnRobertson / TaskManagementSystemTest · 3 days ago Connect
- AlexanderJohnRobertson / dockerTest · 16 days ago Connect
- AlexanderJohnRobertson / responsiveNavbarTest · a month ago Connect

GitHub
@AlexanderJohnRobertson · 17 repos
[Configure account](#)

GitLab
[+ Connect account](#)

Bitbucket
[+ Connect account](#)

Figure 101 - Select the GitHub repository you created in the previous section.

7. Set the Name to an appropriate name for the Task Management System deployment, the Region, Branch of your Git repository, Set the Runtime to “Python 3”. Make sure the Build Command is set to “`pip install -r requirements.txt`” and the Start Command is set to “`gunicorn app:app`”.

You are deploying a web service for [AlexanderJohnRobertson/TaskManagementSystemDeployment](#).

Name
A unique name for your web service. TaskManagementSystemDeployment

Region
The region where your web service runs. Services must be in the same region to communicate privately and you currently have services running in Frankfurt. Frankfurt (EU Central)

Branch
The repository branch used for your web service. main

Root Directory Optional
Defaults to repository root. When you specify a root directory that is different from your repository root, Render runs all your commands in the specified directory and ignores changes outside the directory. e.g., src

Runtime
The runtime for your web service. Python 3

Build Command
This command runs in the root directory of your repository when a new version of your code is pushed, or when you deploy manually. It is typically a script that installs libraries, runs migrations, or compiles resources needed by your app. \$ pip install -r requirements.txt

Start Command
This command runs in the root directory of your app and is responsible for starting its processes. It is typically used to start a webserver for your app. It can access environment variables defined by you in Render. \$ gunicorn app:app

Figure 102 - Render.com web service settings.

8. Set the Instance Type to a subscription dependent on your organisation's needs.

The screenshot shows the 'Instance Type' section of the Render.com interface. It includes a table of instance types with their prices and specifications:

Instance Type	RAM	CPU
Free	512 MB (RAM)	0.1 CPU
Starter \$7 / month	512 MB (RAM)	0.5 CPU
Standard \$25 / month	2 GB (RAM)	1 CPU
Pro \$85 / month	4 GB (RAM)	2 CPU
Pro Plus \$175 / month	8 GB (RAM)	4 CPU
Pro Max \$225 / month	16 GB (RAM)	4 CPU
Pro Ultra \$450 / month	32 GB (RAM)	8 CPU

Below the table, there is a note: "Need a custom instance type? We support up to 512 GB RAM and 64 CPUs." At the bottom, there is a section for "Environment Variables" with an optional note: "Set environment-specific config and secrets (such as API keys), then read those values from your code. Learn more." There are input fields for "NAME_OF_VARIABLE" and "value", and buttons for "Add Environment Variable" and "Add from .env".

Figure 103 - Render.com subscriptions.

9. Click “Create Web Service”.



Figure 104 - Create Web Service

10. Wait for Render.com to build and deploy the Task Management Web Service

The screenshot shows the Render.com TaskManagementSystemDeployment page. It displays the following information:

- WEB SERVICE**: TaskManagementSystemDeployment (Python 3, Free, Upgrade your instance →)
- Events**: Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more. Upgrade now.
- Logs**: Shows a log entry for March 24, 2024 at 5:37 PM: "Building 7268dd6 first commit". A "Cancel deploy" button is present.
- Disks**: Shows a log entry for March 24, 2024 at 5:37 PM: "...Building 7268dd6 first commit".
- Environment**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Running build command 'pip install -r requirements.txt'..."
- Shell**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Collecting Flask (from -r requirements.txt (line 1))".
- Previews**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Downloading Flask-3.0.2-py3-none-any.whl.metadata (3.6 kB)".
- Jobs**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Collecting gunicorn (from -r requirements.txt (line 2))".
- Metrics**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Downloading gunicorn-21.2.0-py3-none-any.whl.metadata (4.1 kB)".
- Scaling**: Shows a log entry for March 24, 2024 at 5:38:11 PM: "...Collecting Werkzeug=3.0.0 (from Flask->r requirements.txt (line 1))".
- Settings**: Shows a log entry for March 24, 2024 at 5:38:13 PM: "...Downloading werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)".
- Logs**: Shows a log entry for March 24, 2024 at 5:38:13 PM: "...Collecting Jinja2>=3.1.2 (from Flask->r requirements.txt (line 1))".
- Logs**: Shows a log entry for March 24, 2024 at 5:38:13 PM: "...Downloading Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)".

Figure 105 - Render.com build progress.

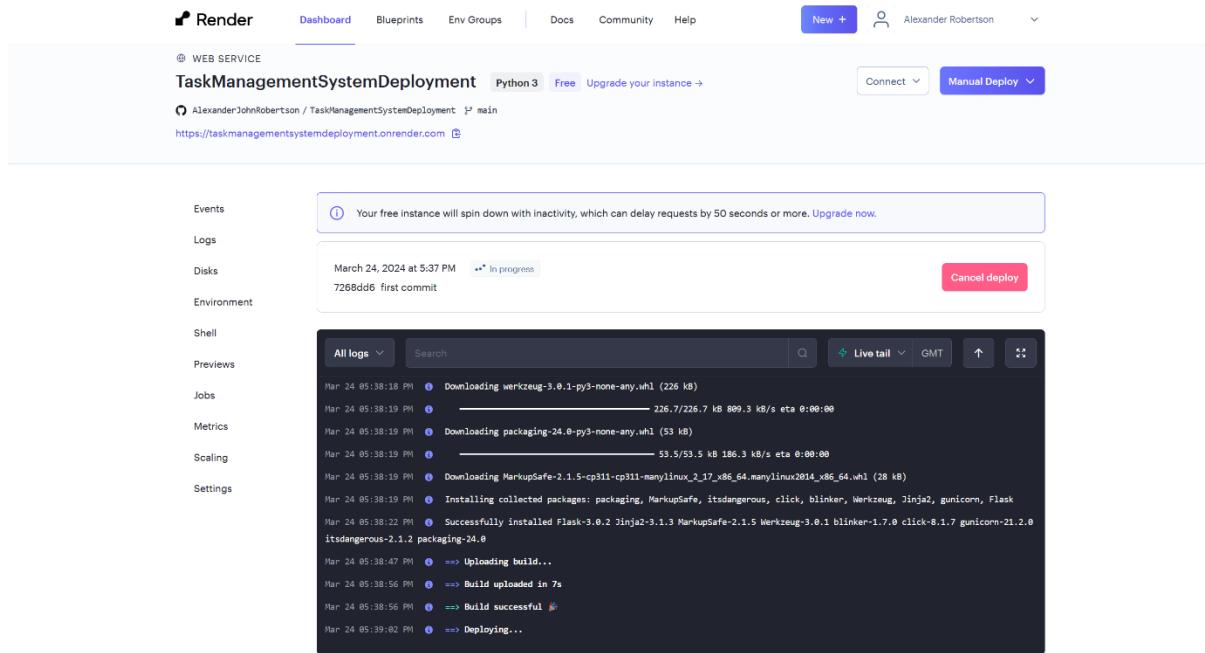


Figure 106 - Render.com deployment progress.

11. The Task Management Web Service deployment is now live. Click the link to view the web service.

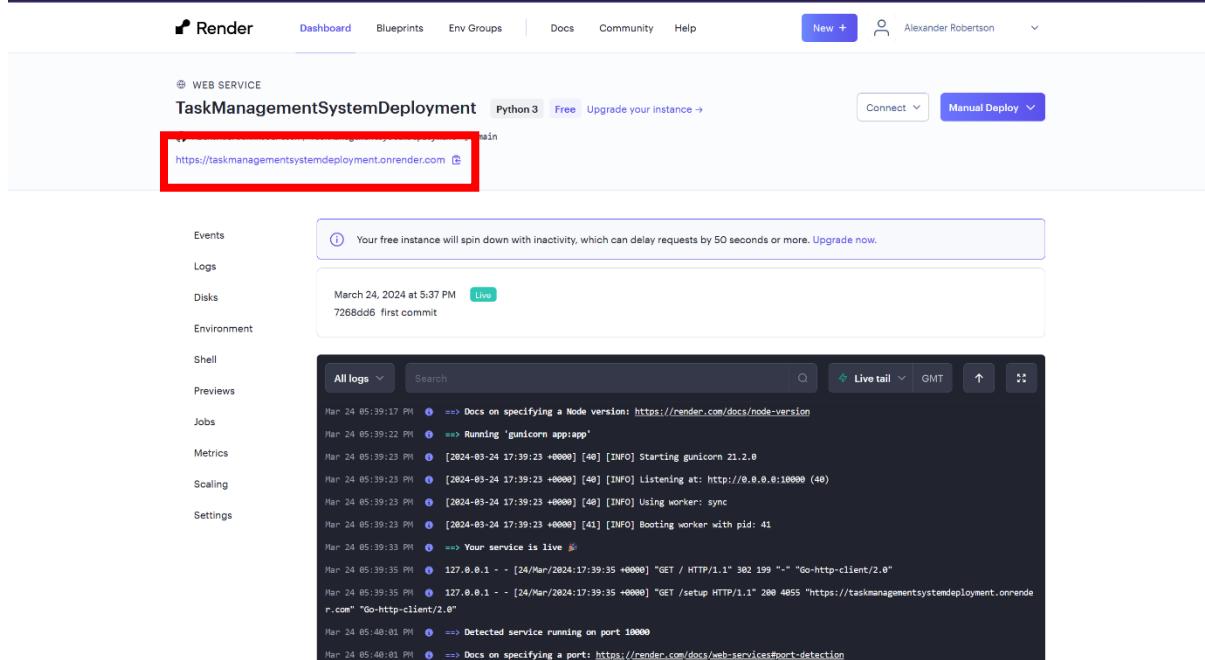


Figure 107 - Build and deploy now complete. Task Management System web service is now live at the given URL.

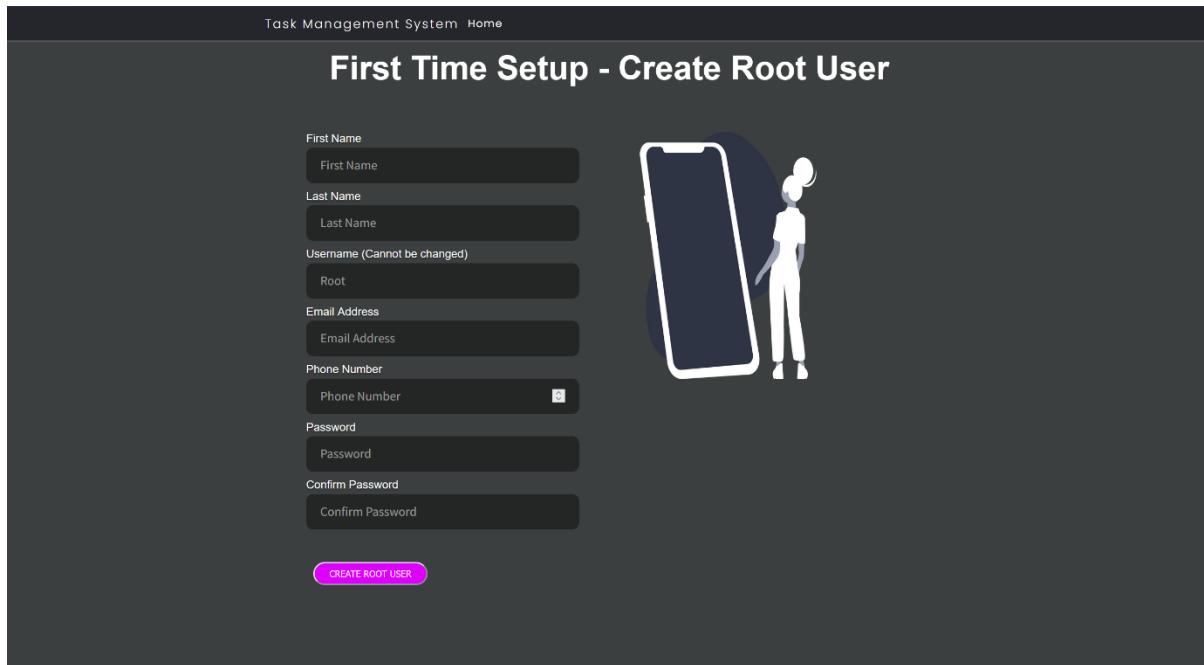


Figure 108 - Task Management System Setup page on Render.com.

12. The Task Management System Web Service is now ready to set up and use.
13. You have now created a Continuous Integration / Continuous Deployment (CI/CD) Pipeline with PyCharm (Code and test (e.g. unit testing)) → Git → GitHub → Render.com (Build, Deploy and Release) → Task Management System as the process is automated as soon as you push to the Git repository from the PyCharm terminal, the Task Management System is deployed on Render.com. Render.com automatically detects new commits and pushes to the Git repository and will then automatically rebuild and redeploy the application.

```

task-management-system-clean-install main
Project task-management-system-clean-install [TaskManagementSystem]
  app.py
    """This is a test page for testing purposes."""
    return render_template('testpage.html')
  @app.route('/testpage2')
  def testpage2():
    """This is a test page for testing purposes."""
    for i in range(10):
      print("Hello World., How are you?")
    return render_template('testpage2.html')
  @app.route('/test3')
  def testpage3():
    """This is a test page for testing purposes."""
    return render_template('test3.html')
  testpage2()
  name == __main__:
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/AlexanderJohnRobertson/TaskManagementSystemDeployment.git
 * [new branch] main -> main
(.venv) PS C:\Users\ruber\newFolder10\task-management-system-clean-install> git add .
(.venv) PS C:\Users\ruber\newFolder10\task-management-system-clean-install> git commit -m "Changed some code"
[main d23b93b] Changed some code
 1 file changed, 1 insertion(+), 1 deletion(-)
(.venv) PS C:\Users\ruber\newFolder10\task-management-system-clean-install> git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 32 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/AlexanderJohnRobertson/TaskManagementSystemDeployment.git
 * [new branch] main -> main
(.venv) PS C:\Users\ruber\newFolder10\task-management-system-clean-install>

```

The screenshot shows a PyCharm interface with a project named 'task-management-system-clean-install'. The 'app.py' file is open, displaying code for three routes: '/testpage2', '/testpage3', and a main route. The main route uses a for loop to print 'Hello World., How are you?' 10 times and returns 'testpage2.html'. The other two routes return 'testpage2.html' and 'test3.html' respectively. Below the code editor is a terminal window. It shows the user navigating to the project directory, running 'git add .' to stage changes, committing with the message 'Changed some code', and pushing the changes to the 'main' branch on GitHub. The commit hash 'd23b93b' is shown. The terminal also shows the remote resolution of deltas and the final push status.

Figure 109 - Committing and pushing new changes to the Git repository.

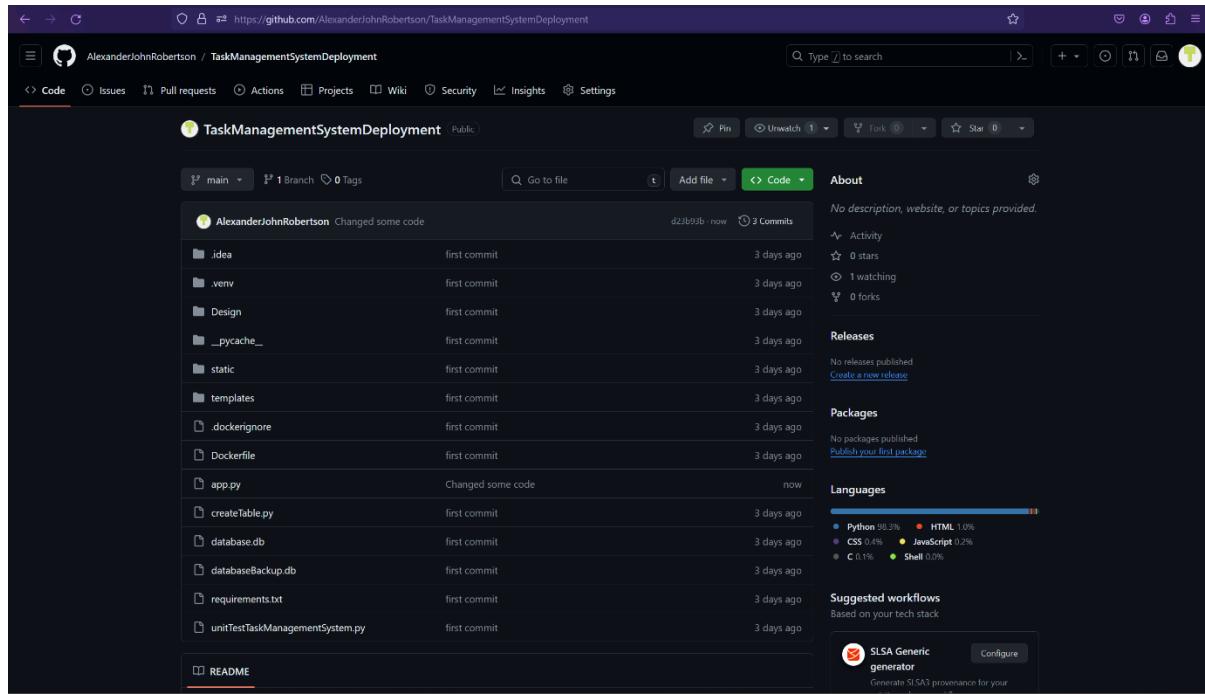


Figure 110- New changes committed to Git repository.

A screenshot of the Render dashboard showing the 'Overview' section. The dashboard lists several services:

Service Name	Status	Type	Runtime	Region	Last Deployed
TaskManagementSystemDeployment	Deploying	Web Service	Python 3	Frankfurt	29 minutes ago
Task Management System (..)	Deployed	Web Service	Python 3	Frankfurt	3 days ago
Task-Management-System	Deployed	Web Service	Python 3	Frankfurt	3 days ago
TaskManagementSystemTest	Deployed	Web Service	Python 3	Frankfurt	3 days ago
Responsive-Navbar-Test	Deployed	Web Service	Python 3	Frankfurt	a month ago
flask-test-4	Deployed	Web Service	Python 3	Frankfurt	2 months ago
flasktest2	Deployed	Web Service	Python 3	Frankfurt	2 months ago
flasktest	Failed deploy	Static Site	Static	Global	2 months ago

Figure 111 - Task Management System web service redeploying shown on Dashboard.

S275931

The screenshot shows the Render.com dashboard for a Python 3 web service named 'TaskManagementSystemDeployment'. The 'Events' tab is selected, displaying a list of deployment logs. A specific log entry from March 24, 2024, at 6:26 PM is highlighted with a red box:

```
Deploy started for 515e396: Changed some code
New commit via Auto-Deploy
March 24, 2024 at 6:26 PM
```

Figure 112 - Task Management System redeployment showing in web service "Events" tab.

The screenshot shows the Render.com dashboard for the same web service. The 'Logs' tab is selected, displaying a log stream. A specific entry from March 24, 2024, at 6:26 PM is highlighted with a red box:

```
Mar 24 06:26:57 PM *** Building
515e396: Changed some code
Cancel deploy
```

Below this, the log stream continues with several cached package installations and a successful Flask installation:

```
Mar 24 06:26:57 PM Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Mar 24 06:26:57 PM Using cached Jinja2-3.1.3-py3-none-any.whl (133 kB)
Mar 24 06:26:57 PM Using cached werkzeug-3.0.1-py3-none-any.whl (236 kB)
Mar 24 06:26:57 PM Using cached packaging-24.0-py3-none-any.whl (53 kB)
Mar 24 06:26:57 PM Using cached MarkupSafe-2.1.5-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (28 kB)
Mar 24 06:26:57 PM Installing collected packages: packaging, MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, gunicorn, Flask
Mar 24 06:26:57 PM Successfully installed Flask-3.0.2 Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-2.1.0 click-8.1.7 gunicorn-21.2.0
itsdangerous-2.1.2 packaging-24.0
Mar 24 06:26:59 PM *** Uploading build...
Mar 24 06:26:52 PM *** Transferred 55MB in 7s. Extraction took 1s.
Mar 24 06:27:07 PM *** Build uploaded in 7s
Mar 24 06:27:07 PM Build succeeded
```

Figure 113 - Task Management System web service rebuilding as GitHub and Render.com has set up a CI/CD pipeline.

S275931

The screenshot shows the Render.com dashboard for a project named 'TaskManagementSystemDeployment'. A deployment for branch 'main' is currently in progress, indicated by a red progress bar. The logs panel displays the deployment process, starting with cached dependencies like Jinja2 and Werkzeug, followed by installing collected packages such as Flask, MarkupSafe, and Click. The logs also show the build being uploaded and successfully deployed, with the final message 'Deploying...'. A 'Cancel deploy' button is visible in the top right of the logs panel.

Figure 114 - Task Management System web service redeploying as GitHub and Render.com has set up a CI/CD pipeline.

The screenshot shows the Render.com dashboard after the redeployment. The deployment status is now 'Live', indicated by a green button. The logs panel shows the deployment process completed successfully, with messages indicating the service is live and receiving traffic. The URL 'https://taskmanagementsystemdeployment.onrender.com' is highlighted with a red box at the top of the browser window.

Figure 115 - Task Management System has been successfully redeployed and is now live.

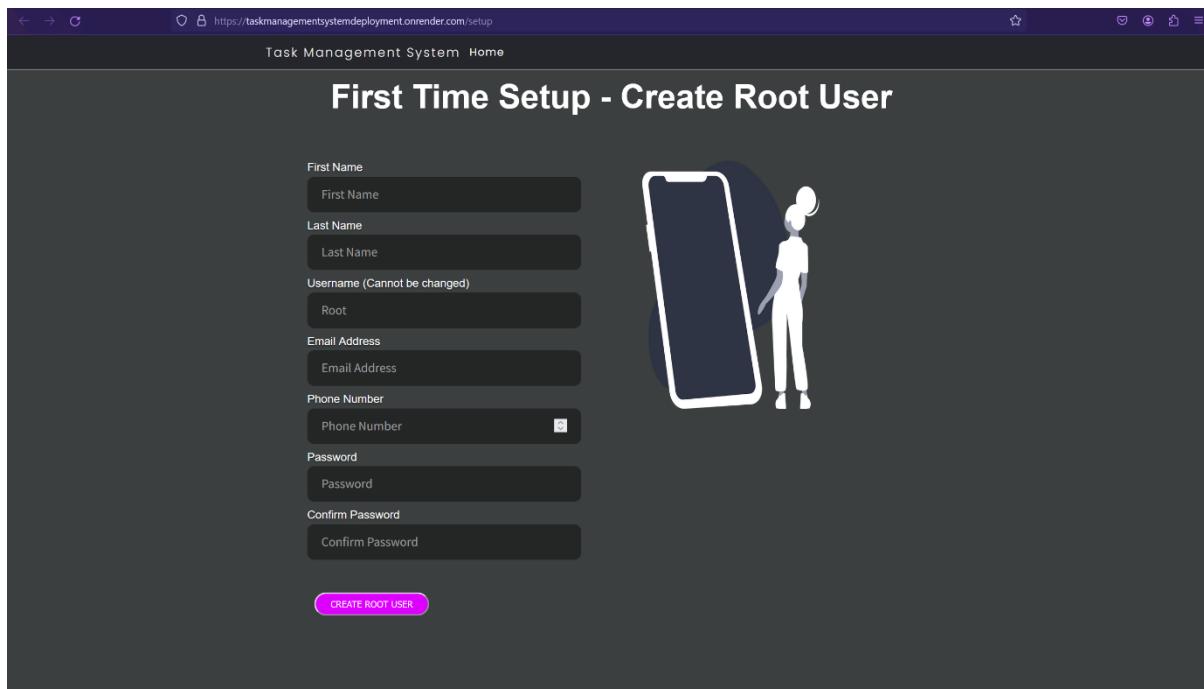


Figure 116 - Task Management System redeployed on Render.com.

Deploy using Docker or other containerisation platforms.

1. Open the folder containing the source code for the Task Management System.

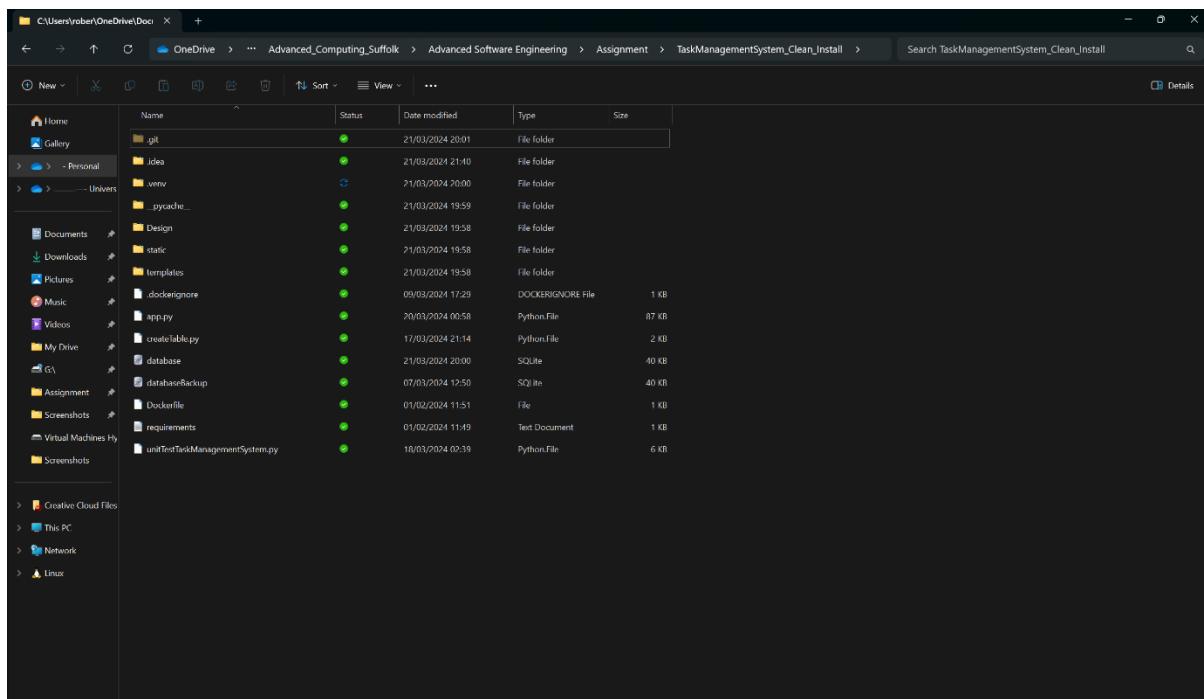


Figure 117 - Directory containing the Task Management System program files.

2. Open the folder containing the Task Management System in the Terminal, PowerShell, or Command Prompt. Alternatively navigate to the folder from within the terminal if your operating system's file browser doesn't support this feature.

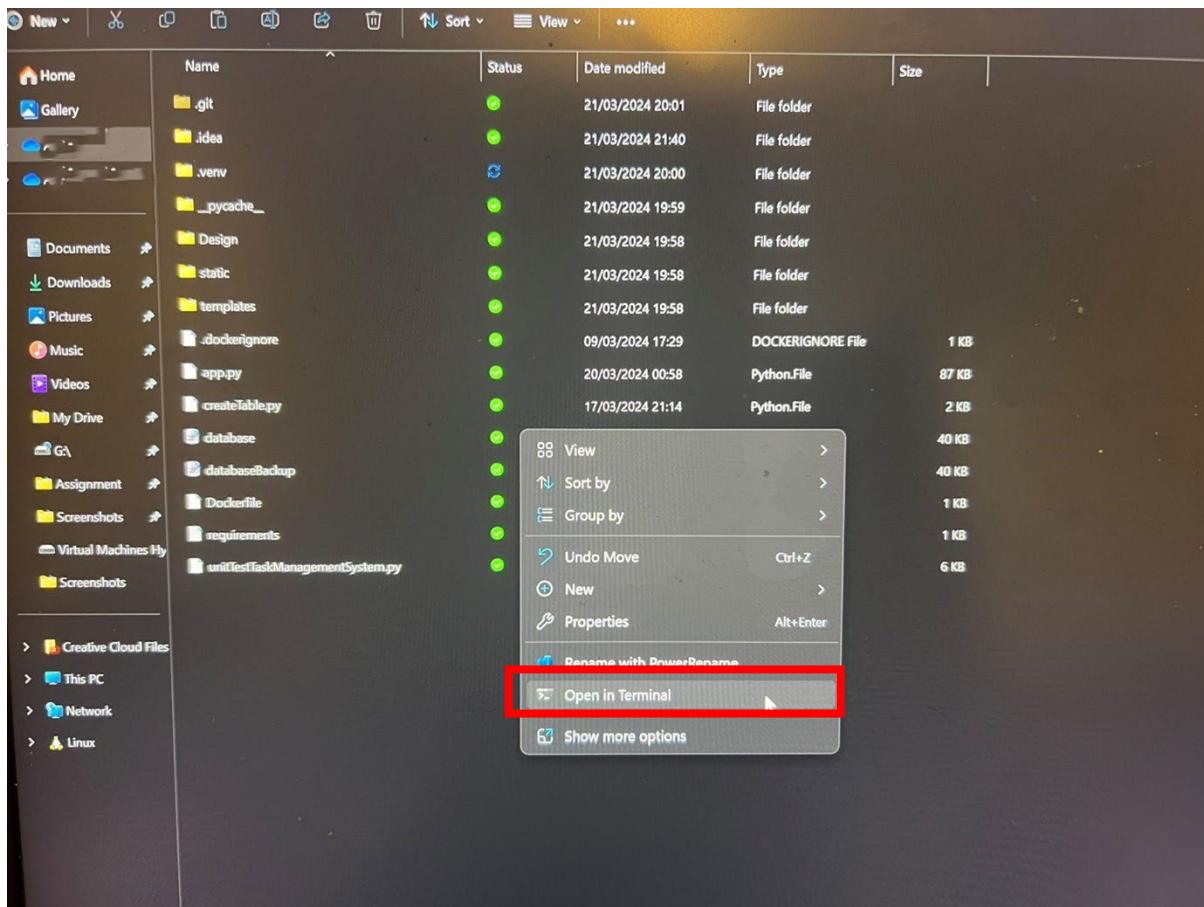


Figure 118 - Opening the Terminal in the Task Management System directory.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

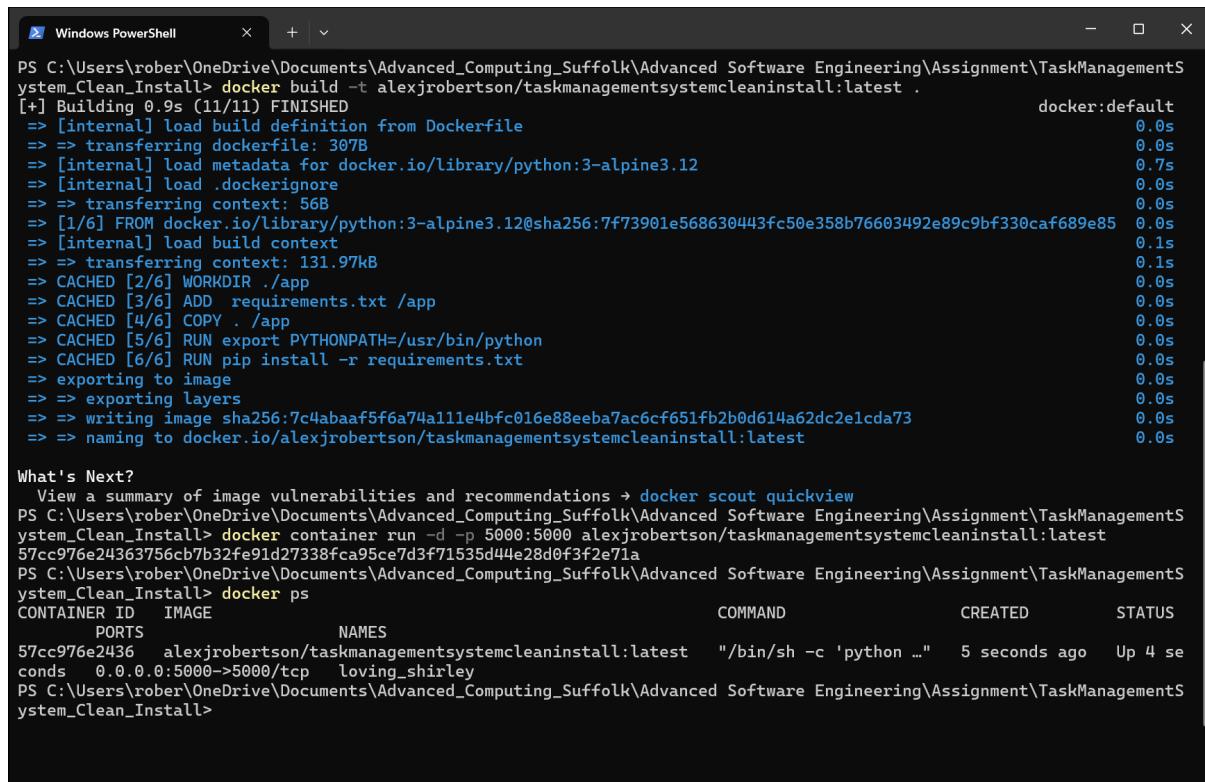
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem_Clean_Install>
```

Figure 119 - The Terminal open in the Task Management System directory.

3. Build the Docker image by entering the command “`docker build -t <your-docker-username>/taskmanagementsystemCleaninstall:latest .`” or `docker build -t <your-docker-username>/taskmanagementsystem:Latest .`”.

4. Run the Docker container by entering the command “`docker container run -d -p 5000:5000 <your-docker-username>/taskmanagementsystemcleaninstall:latest`” or “`docker container run -d -p 5000:5000 <your-docker-username>/taskmanagementsystemcleaninstall:latest`”.
5. Use the command “`docker ps`” to verify that the container is running.



```

PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem_Clean_Install> docker build -t alexjrobertson/taskmanagementsystemcleaninstall:latest .
[+] Building 0.9s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 307B
=> [internal] load metadata for docker.io/library/python:3-alpine3.12
=> [internal] load .dockerrignore
=> => transferring context: 56B
=> [1/6] FROM docker.io/library/python:3-alpine3.12@sha256:7f73901e568630443fc50e358b76603492e89c9bf330caf689e85
=> [internal] load build context
=> => transferring context: 131.97kB
=> CACHED [2/6] WORKDIR ./app
=> CACHED [3/6] ADD requirements.txt /app
=> CACHED [4/6] COPY . /app
=> CACHED [5/6] RUN export PYTHONPATH=/usr/bin/python
=> CACHED [6/6] RUN pip install -r requirements.txt
=> exporting to image
=> => exporting layers
=> => writing image sha256:7c4abaaf5f6a74a11e4bfc016e88eba7ac6cf651fb2b0d614a62dc2e1cda73
=> => naming to docker.io/alexjrobertson/taskmanagementsystemcleaninstall:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem_Clean_Install> docker container run -d -p 5000:5000 alexjrobertson/taskmanagementsystemcleaninstall:latest
57cc976e24363756cb7b32fe91d27338fca95ce7d3f71535d44e28d0f3f2e71a
PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem_Clean_Install> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
57cc976e2436 alexjrobertson/taskmanagementsystemcleaninstall:latest "/bin/sh -c 'python ...'" 5 seconds ago Up 4 seconds 0.0.0.0:5000->5000/tcp loving_shirley
PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem_Clean_Install>

```

Figure 120 - Running the docker build and docker run commands to build the Docker Image and run the Docker Container.

6. View the running container(s) in the Containers tab in Docker Desktop.

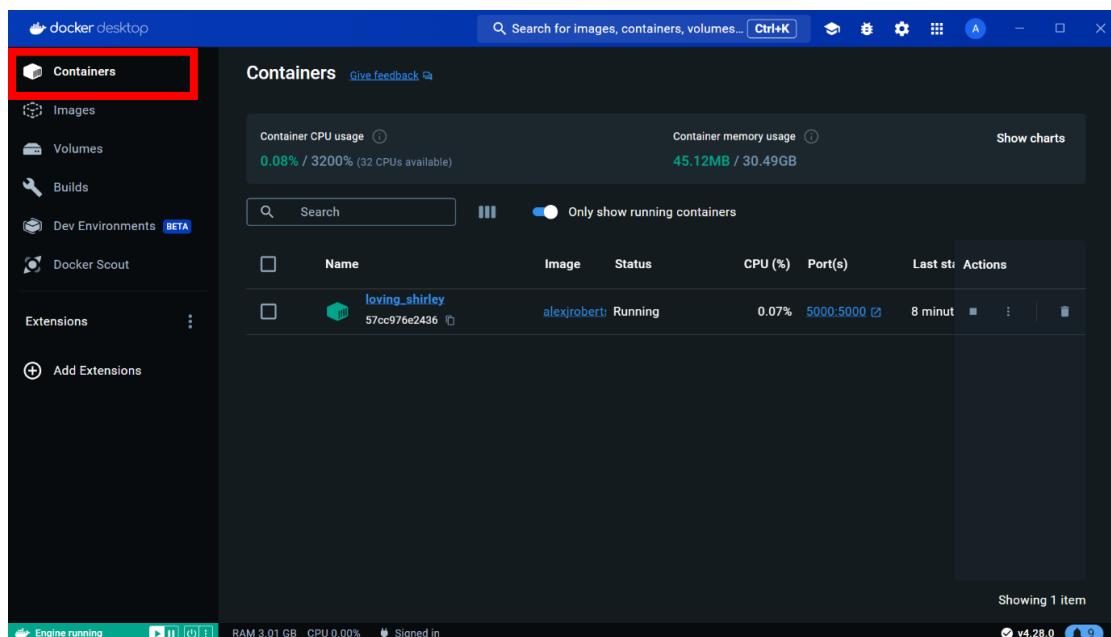


Figure 121 - Task Management System Container running on Docker Desktop.

7. Click on the Port(s) to view the running Task Management System container in your web browser.

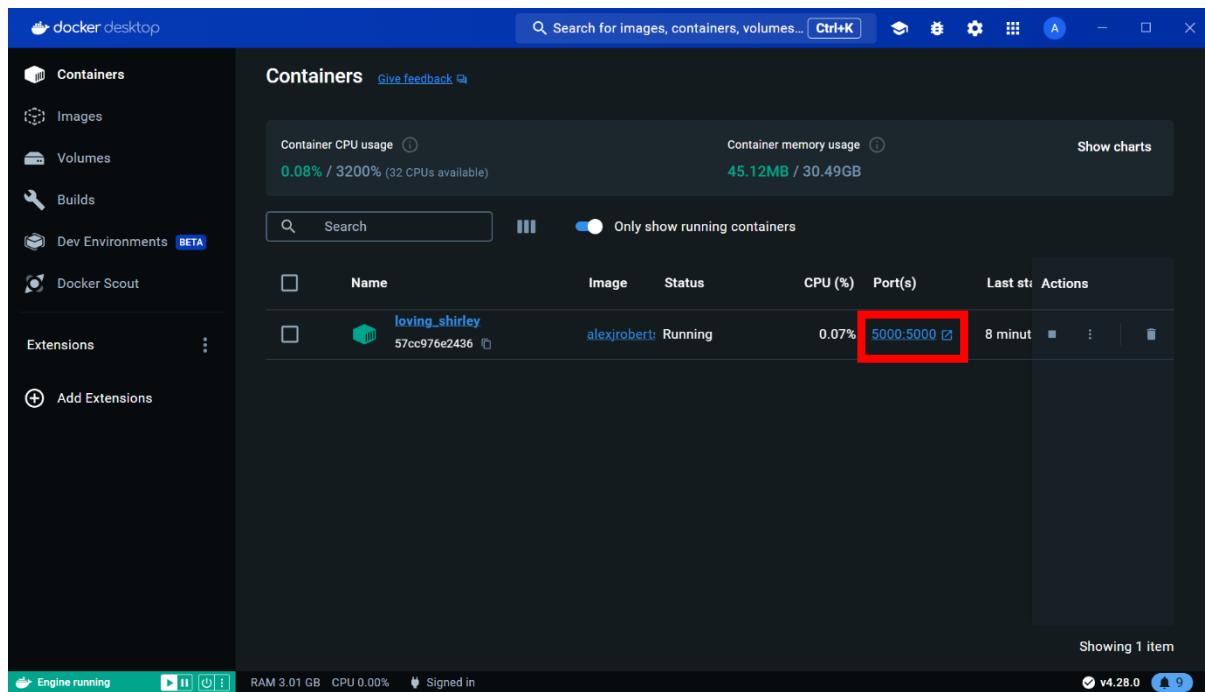


Figure 122 - Select the Port(s) to access the Task Management System running in your browser.

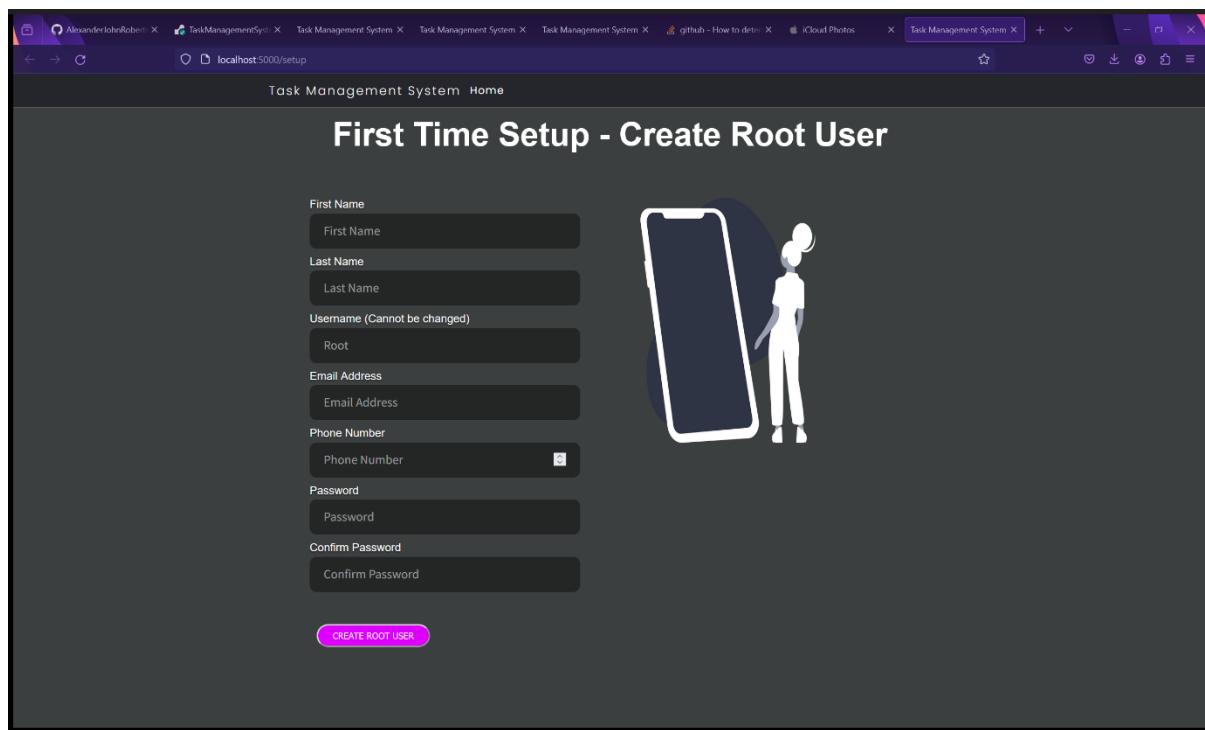


Figure 123 - Task Management System running in a Docker container.

8. View the images on the Images tab in Docker Desktop.

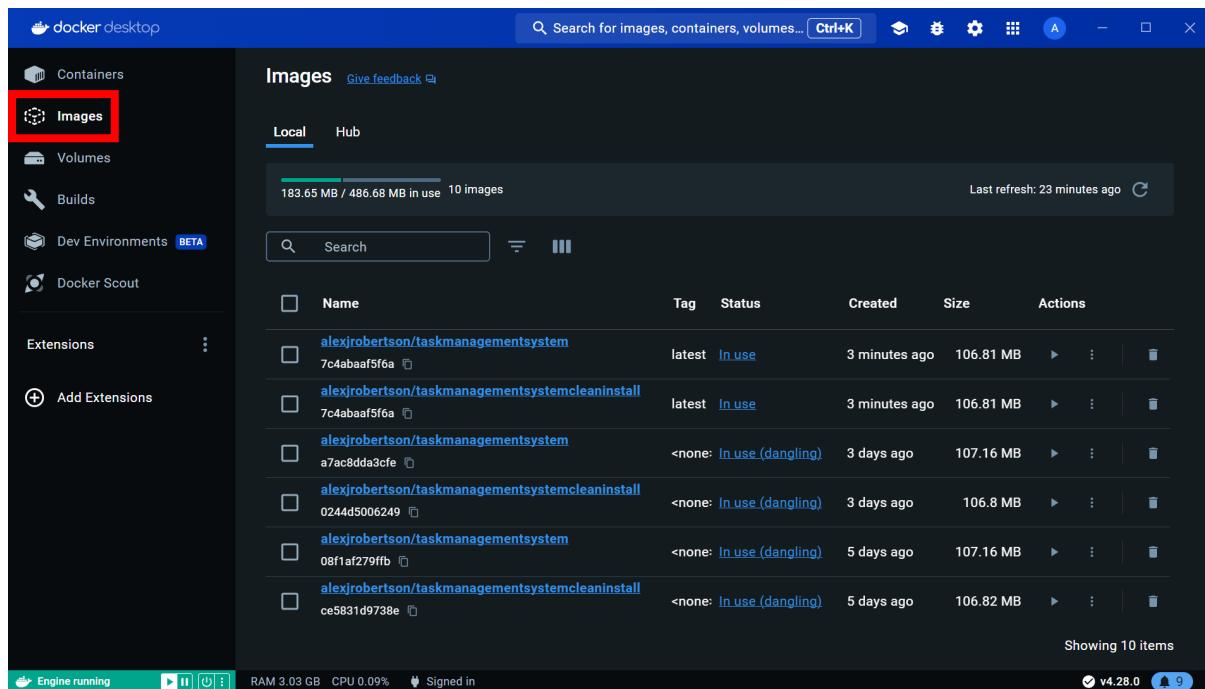


Figure 124 - List of Docker Images.

9. Push the image to Docker Hub.

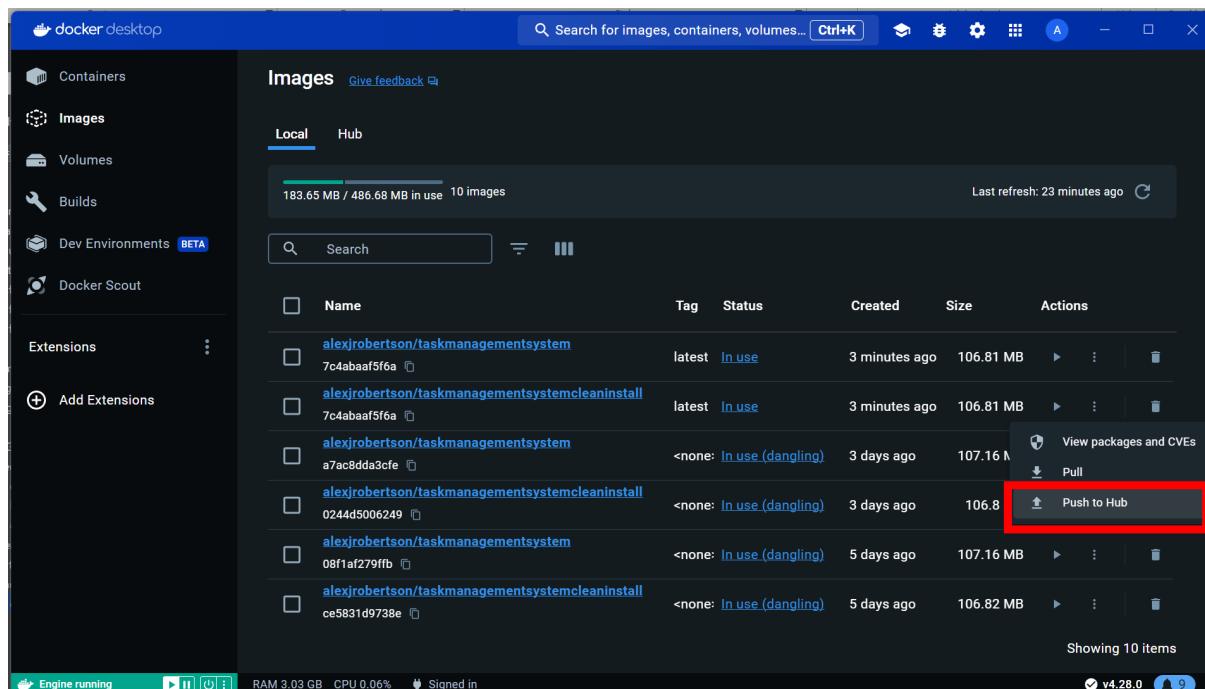


Figure 125 - Pushing the Docker Images to Docker Hub.

10. View the image in Docker Hub.

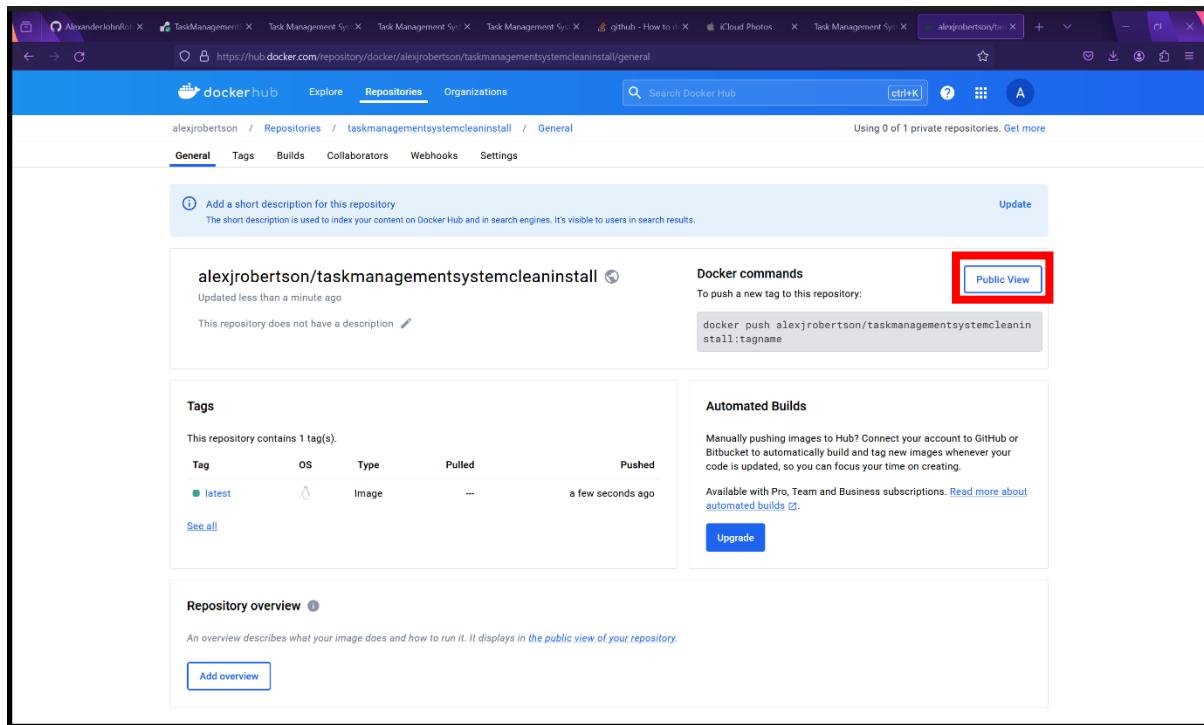


Figure 126 - Task Management System on Docker Hub.

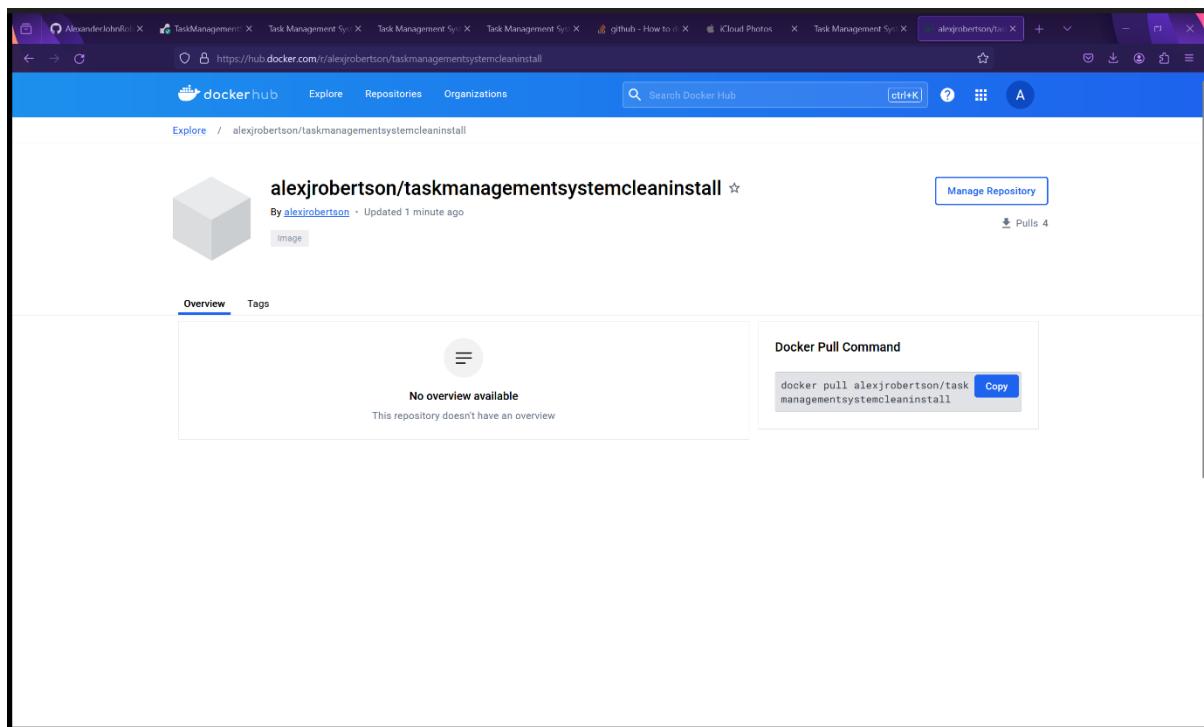
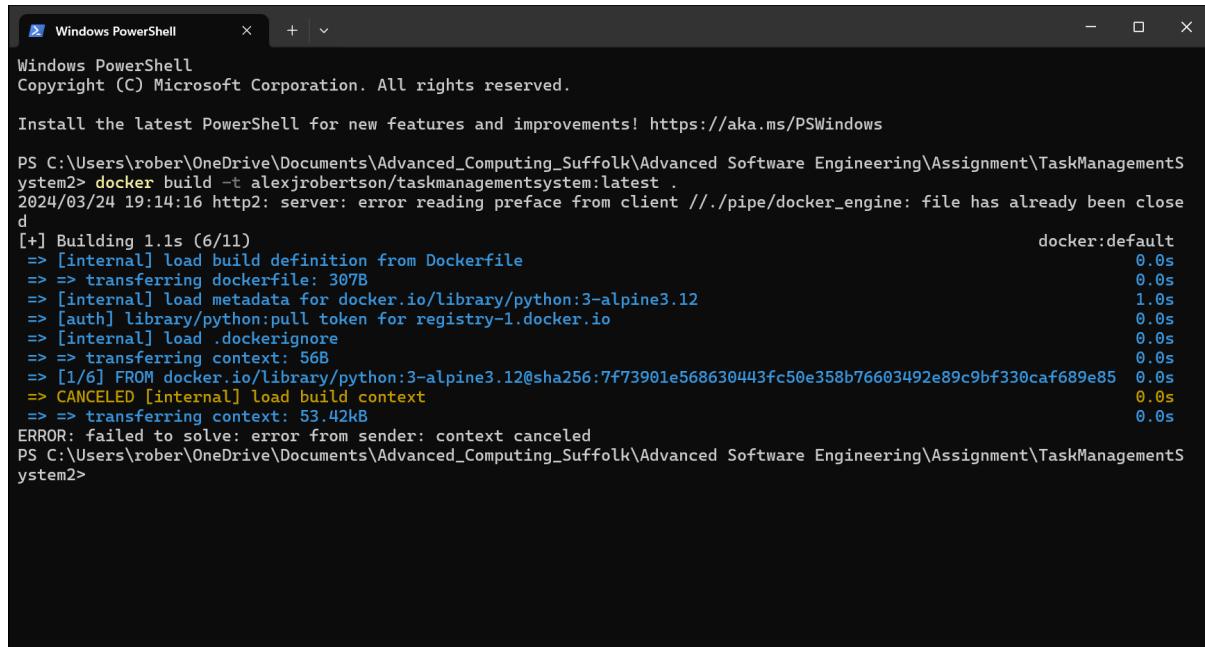


Figure 127 - Public view of the Docker Image on Docker Hub.

11. Troubleshooting tips:

- Make sure there is one space and a full stop (.) at the end of the Docker build command or else the build will fail.
- If the build fails and you encounter an error message in the terminal (*ERROR: failed to solve: error from sender: context canceled*) when entering the docker build command, consider closing PyCharm and re-run the docker

build command. Use the computer terminal instead of the PyCharm terminal. There is a bug that sometimes causes PyCharm to interfere with Docker.



```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2> docker build -t alexjrobertson/taskmanagementsystem:latest .
2024/03/24 19:14:16 http2: server: error reading preface from client //./pipe/docker_engine: file has already been closed
[+] Building 1.1s (6/11)
=> [internal] load build definition from Dockerfile                               docker:default
=> => transferring dockerfile: 307B                                         0.0s
=> [internal] load metadata for docker.io/library/python:3-alpine3.12          0.0s
=> [auth] library/python:pull token for registry-1.docker.io                   1.0s
=> [internal] load .dockerignore                                              0.0s
=> => transferring context: 56B                                         0.0s
=> [1/6] FROM docker.io/library/python:3-alpine3.12@sha256:7f73901e568630443fc50e358b76603492e89c9bf330caf689e85  0.0s
=> CANCELED [internal] load build context                                     0.0s
=> => transferring context: 53.42kB                                         0.0s
ERROR: failed to solve: error from sender: context canceled
PS C:\Users\rober\OneDrive\Documents\Advanced_Computing_Suffolk\Advanced Software Engineering\Assignment\TaskManagementSystem2>

```

Figure 128 - Error that occurs if you try to build the Docker images while PyCharm is still running.

- c. If you receive an error message that the port is already in use when you enter the docker container run command, then change the port number (in red) before the colon and re-run the docker container run command. E.g. change “`docker container run -d -p 5000:5000 <your-docker-username>/taskmanagementsystemcleaninstall:latest`” to “`docker container run -d -p 5100:5000 <your-docker-username>/taskmanagementsystemcleaninstall:latest`”.

Using the Task Management System

First Time Setup

1. On the deployment platform of choice such as Render.com or Docker, open the link to the Task Management System (See Installation instructions above for details).

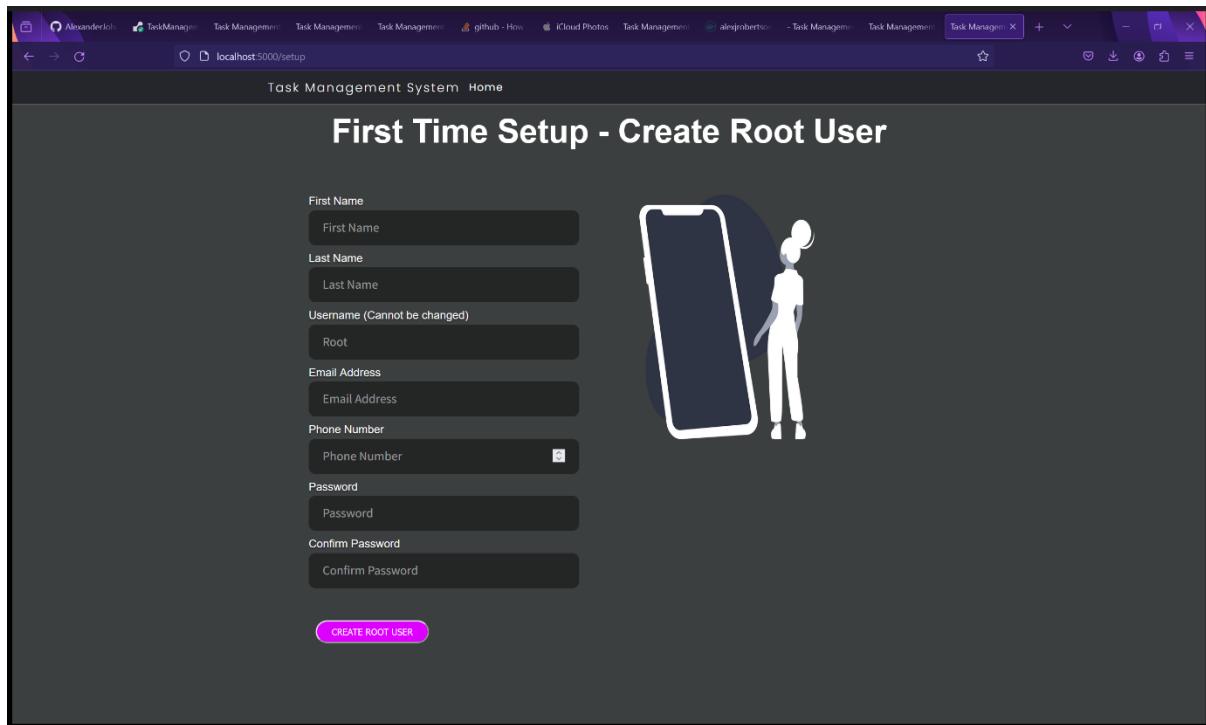


Figure 129 - Task Management System Setup page.

2. Fill in the First Time Setup form to create a Root user. Please note that the username is “Root” and cannot be changed for this user. All fields must be filled in with valid information. The password must be at least 10 characters long, have at least one uppercase letter, one lowercase letter, one number and one special character. The special character(s) must be an ASCII character due to technical limitations with the emailing function. Then click “Create Root User”. The password must be a common password.

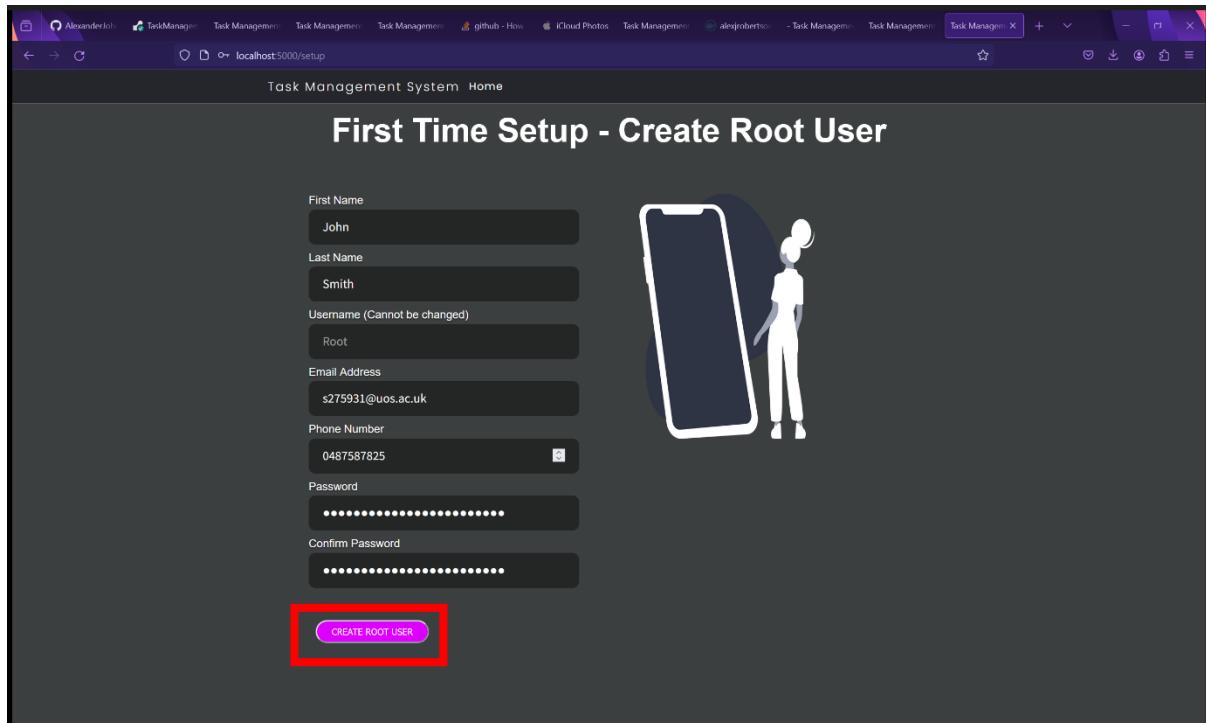


Figure 130 - Task Management System Setup page.

3. You will be redirected to the User home page and receive an email with the password you entered.

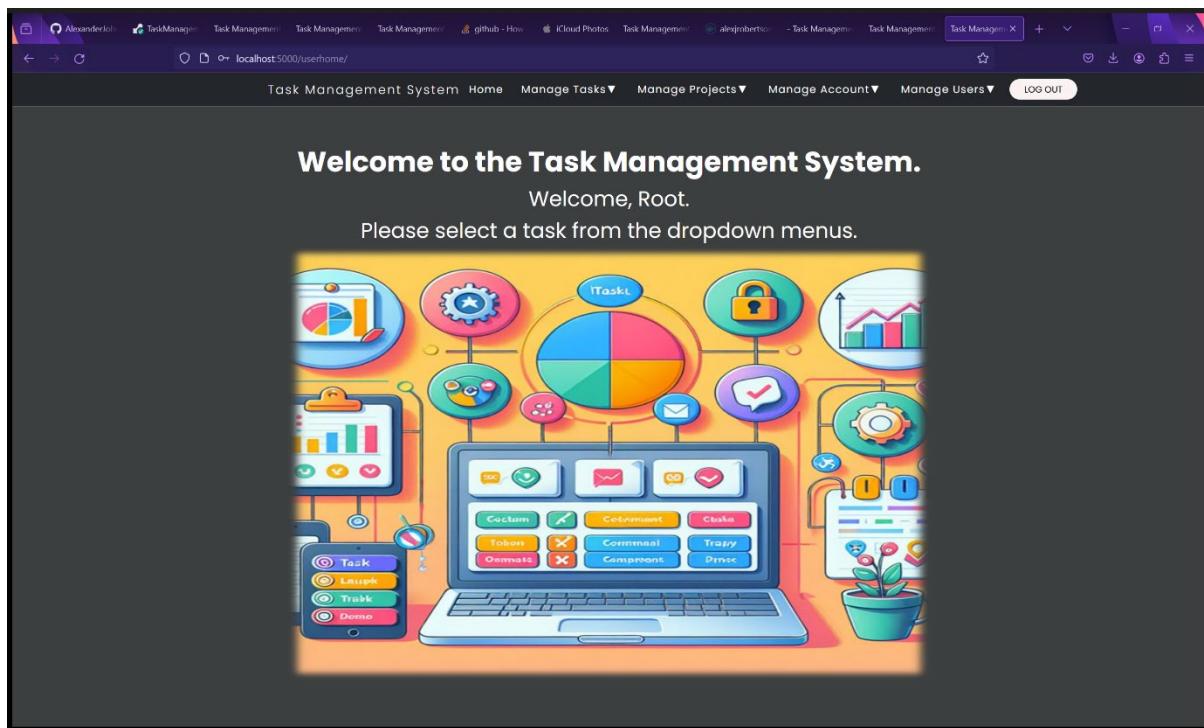


Figure 131 - User Home Page

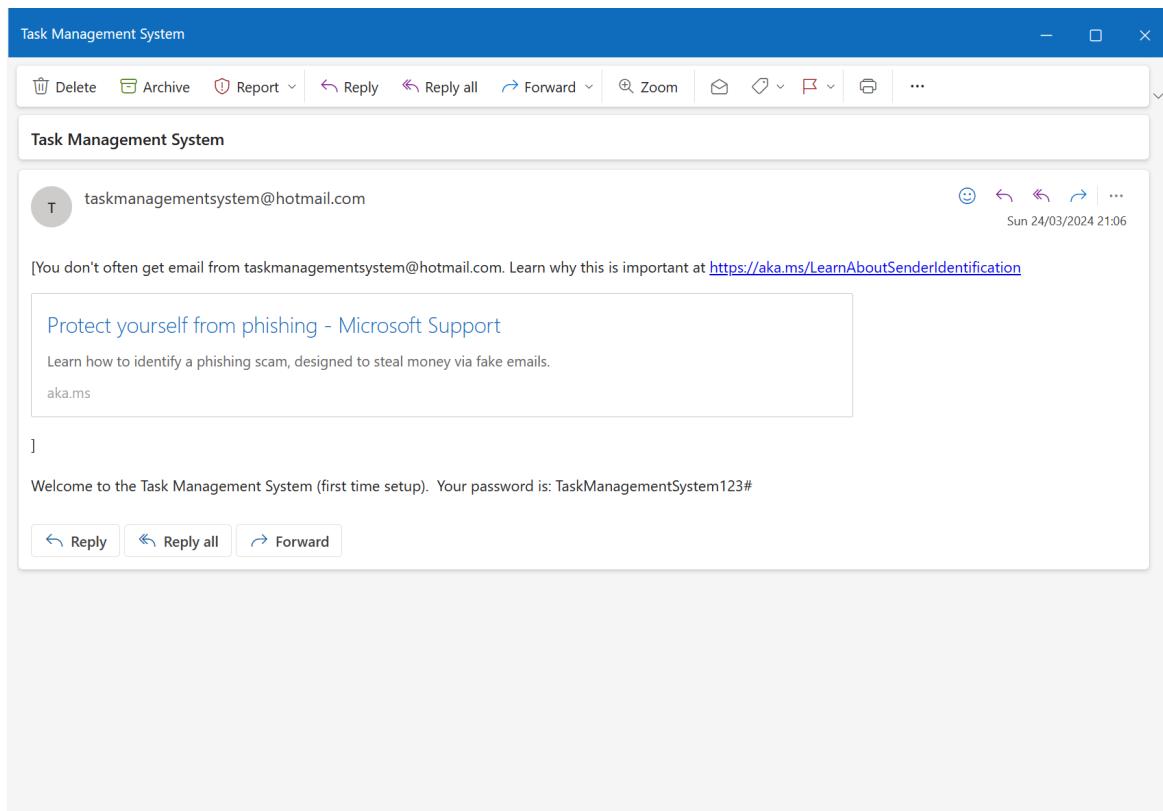


Figure 132 - Automatic email sent to Root User upon setup.

4. The Task Management System is now ready for use.

Logging out

To log out, simply click the "Log Out" button in the top right corner of the navbar. You will be redirected to the Task Management System landing page.

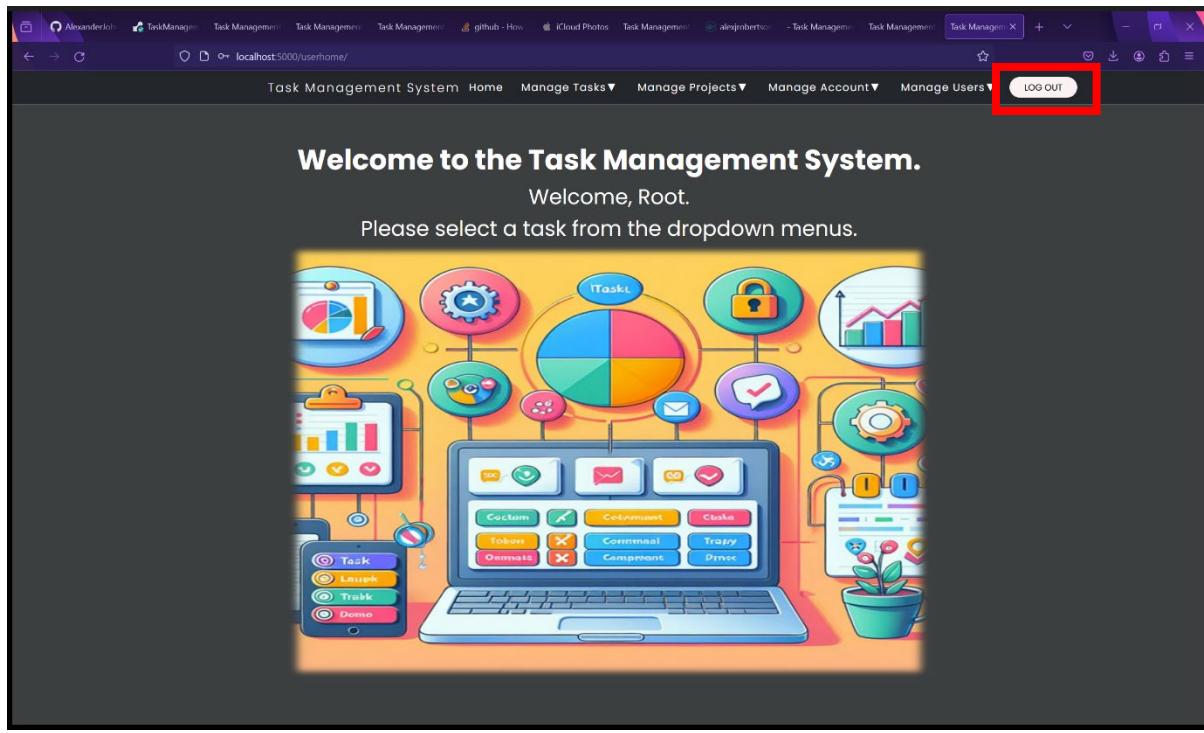


Figure 133 - User Home Page while logged in.

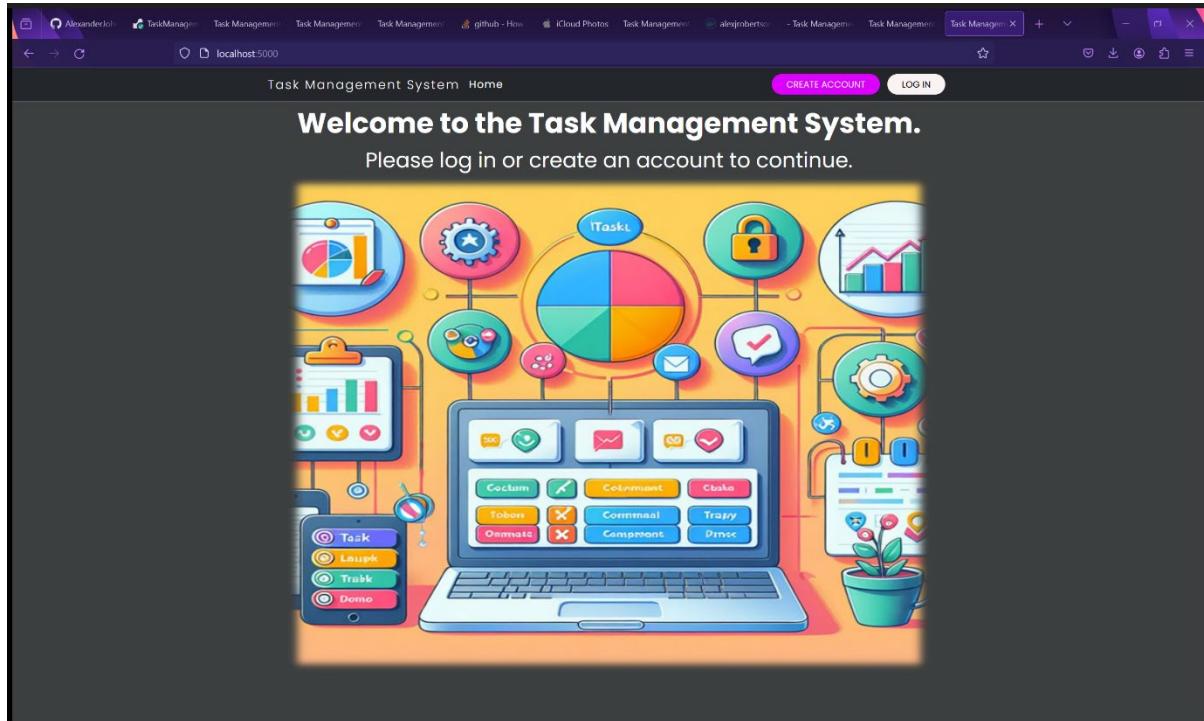


Figure 134 - Landing page after being logged out.

Logging In

1. Click the “Log In” button in the navbar in the top right corner.

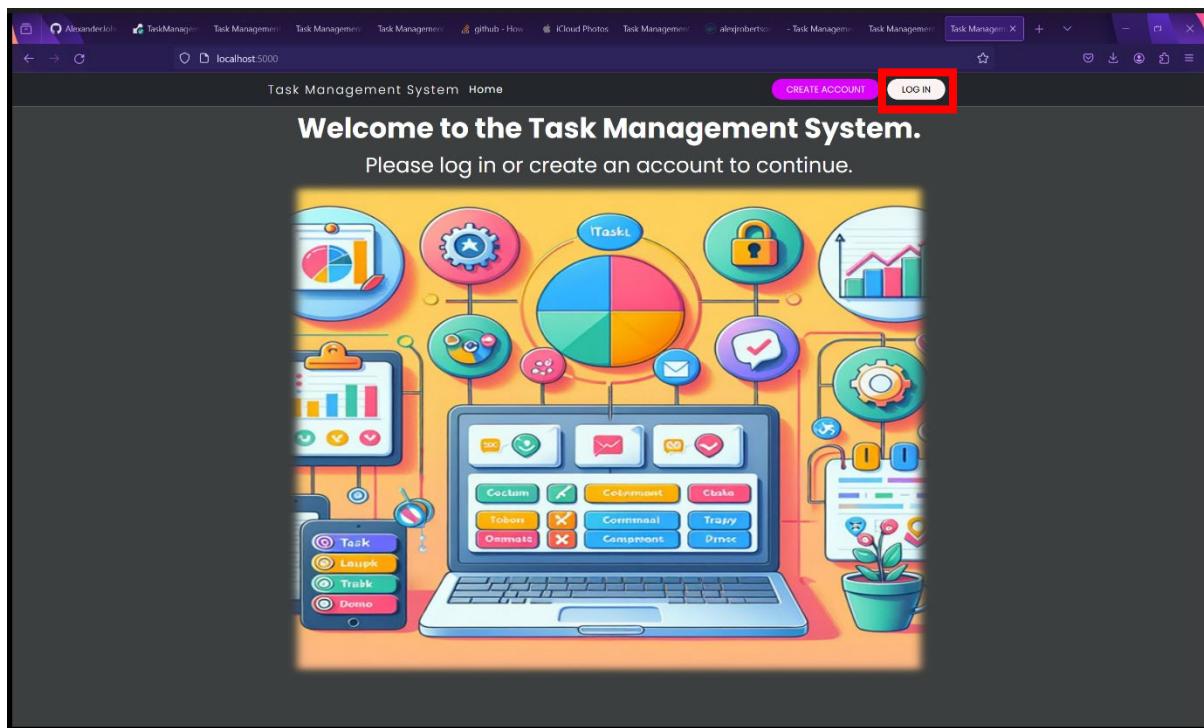


Figure 135 - Landing Page

2. Enter your username and password and click “Log In”. **WARNING: If you enter the wrong password three or more times in a row, your account will be automatically blocked for security reasons. You will need to contact an administrator or Root user to unblock your account.**

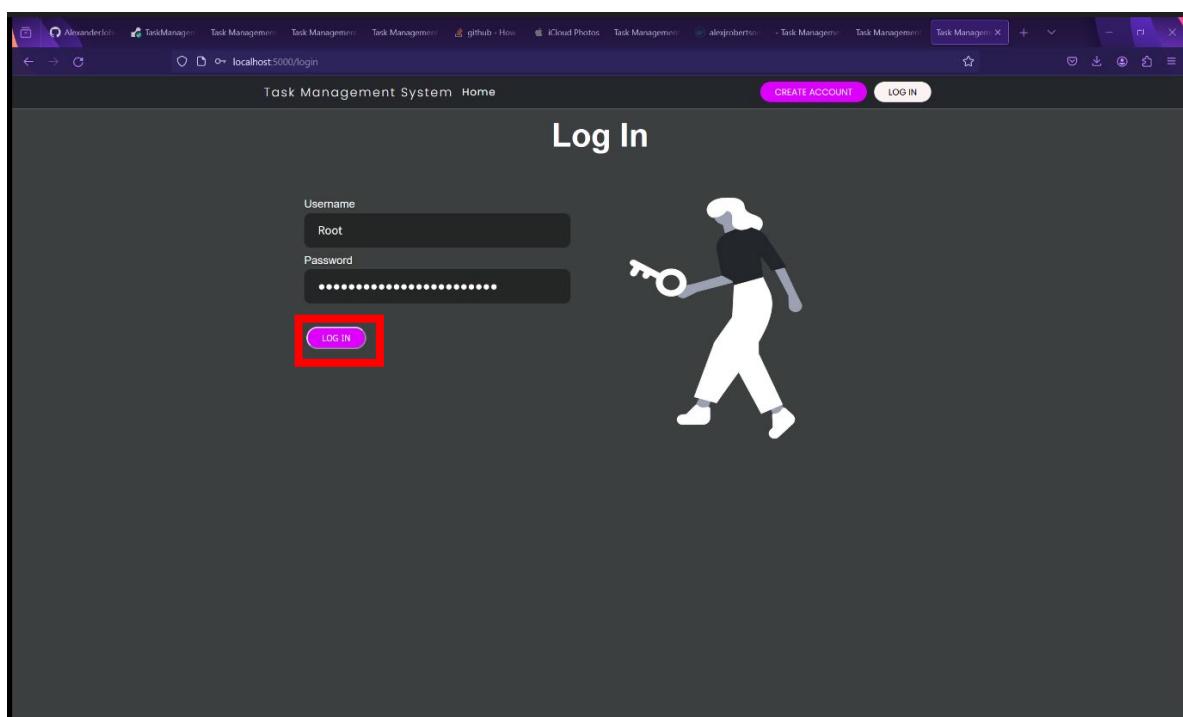


Figure 136 - Log In form.

3. You will be redirected to the User home page.

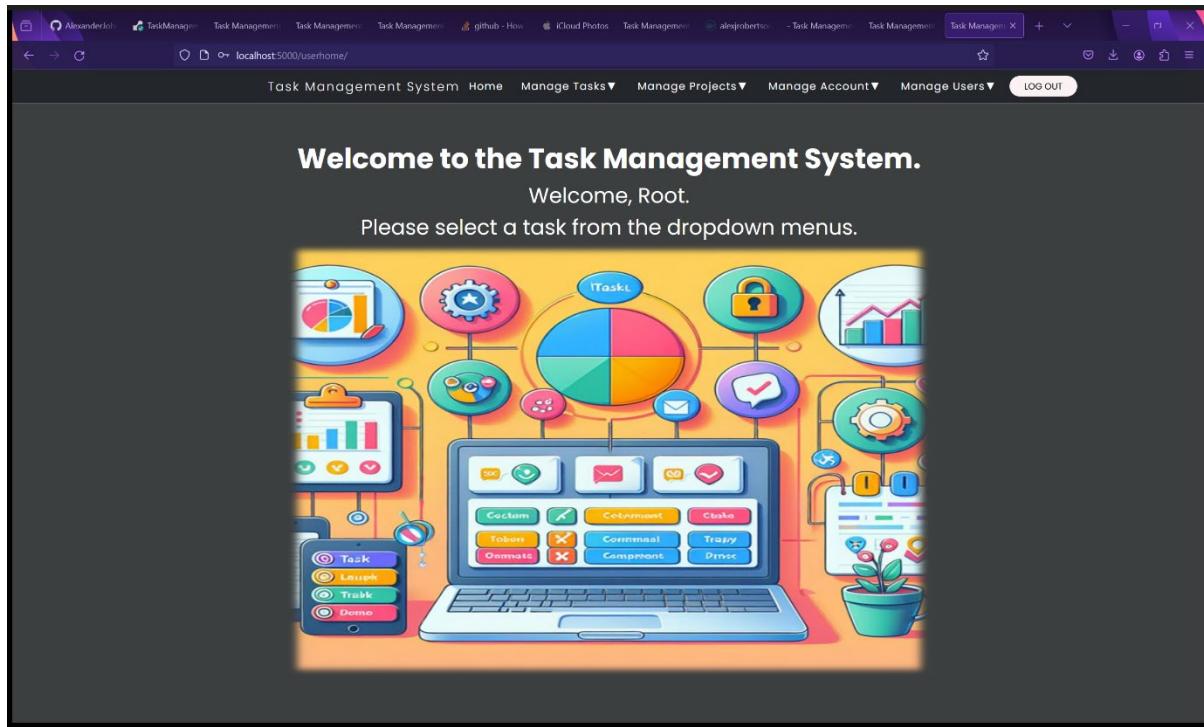


Figure 137 - User Homepage after logging in.

4. You are now able to use the Task Management System.

Create Account

1. Click the “Create Account” button in the navbar in the top right corner.

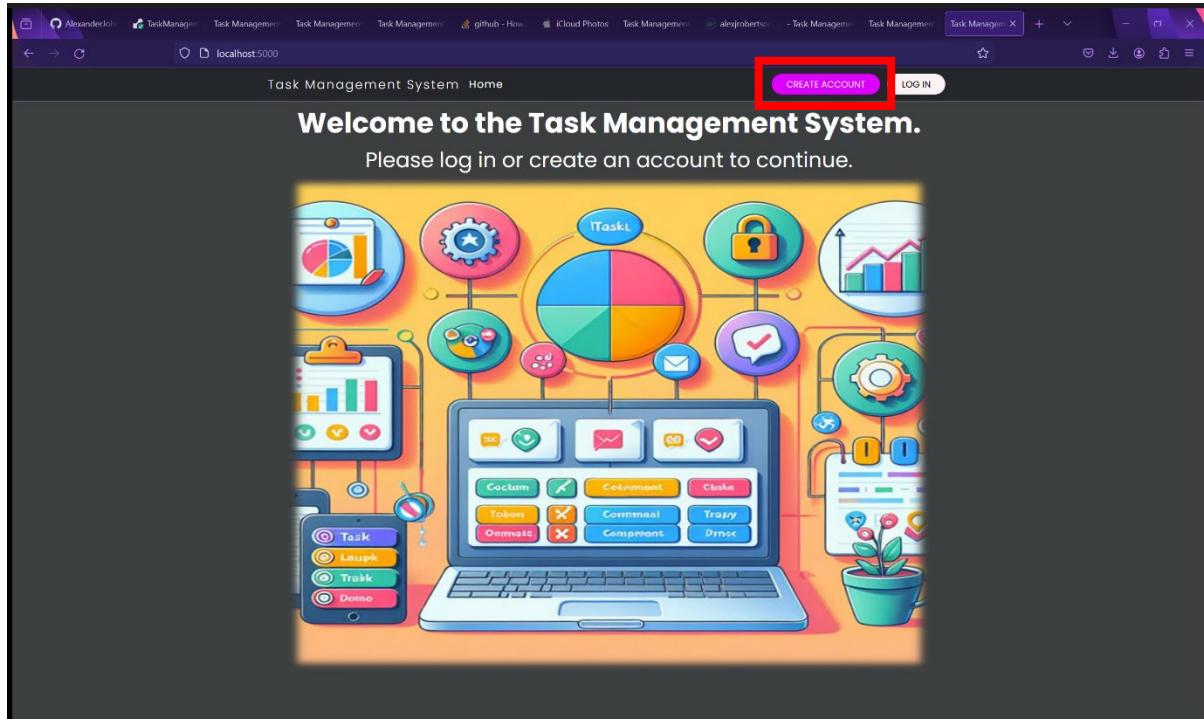


Figure 138 - Landing Page

2. Fill in the Create Account form with your details then click the “Create Account” button. Please note that the password must be at least 10 characters long, have at least one uppercase letter, one lowercase letter, one number and one special character and must not be a common password. Non-ASCII characters such as UTF and Unicode characters are allowed as an email is not automatically sent in this case.

The screenshot shows a web browser window with the URL `localhost:5000/createaccount`. The page title is "Task Management System Home". On the left, there is a sidebar with several links: "Task Management", "Task Management", "Task Management", "github - How", "iCloud Photos", "Task Management", "alexjrobertson - Task Management", "Task Management", and "Task Management". On the right, there are standard browser controls: back, forward, search, and refresh. Below the title, there is a "CREATE ACCOUNT" button and a "LOG IN" button. The main content area is titled "Create Account". It contains the following form fields:

First Name	Adam
Last Name	McCann
Username	adammccann12
Email Address	robertson97@hotmail.co.uk
Phone Number	07775385490
Password	*****
Confirm Password	*****

At the bottom of the form, there is a "CREATE ACCOUNT" button, which is highlighted with a red rectangular box.

Figure 139 - Account Creation Form.

3. The user is redirected to their user home page and are now ready to use the Task Management System. Please note that new user accounts are created as a standard user by default for security purposes therefore there are restrictions on standard accounts. A standard account can add and view tasks, add and view projects, view and change their own account details except for their account type and block status. An administrator account and the Root user both have full control over the Task Management System and therefore can add, view, update and delete tasks, add, view, update and delete projects, manage other users, and view other user details and perform administrative tasks. Administrators and the Root user can change the account type of other users, block and unblock users and factory reset the Task Management System.

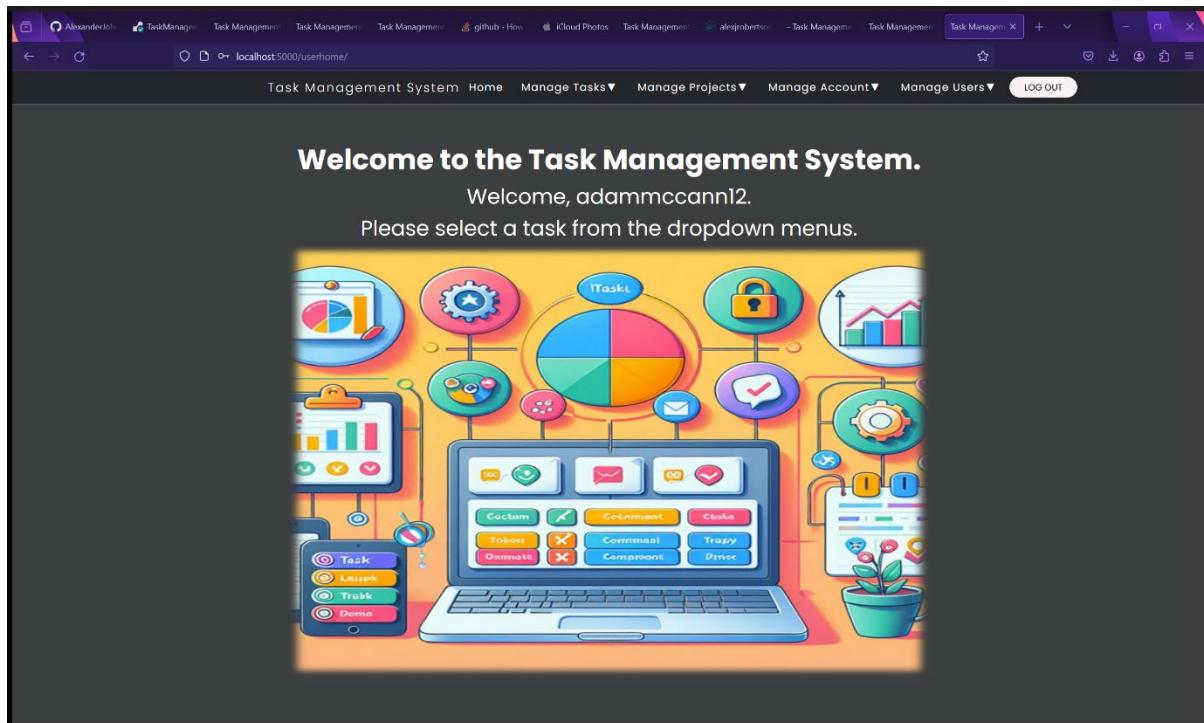


Figure 140 - User Home Page after Account Creation.

4. The Task Management System is ready for use.

Manage Projects

Create a New Project

1. Select “Create Project” from the “Manage Projects” dropdown menu.

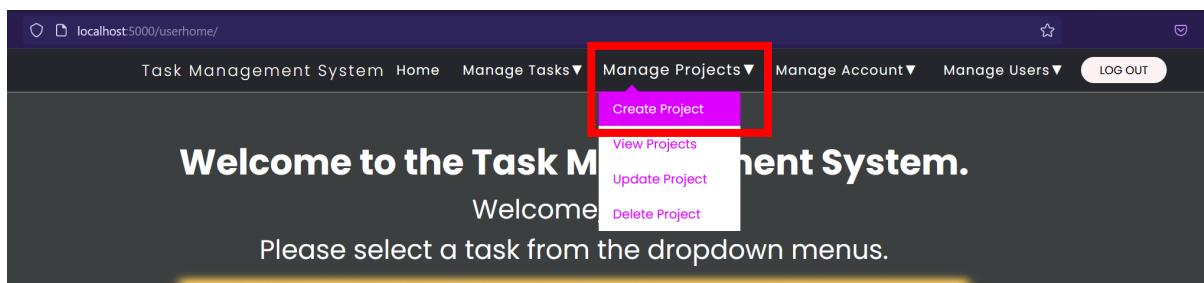


Figure 141 - Select “Create Project” from “Manage Projects” dropdown menu.

2. You will be redirected to the Add New Project form.
3. Fill in the details of the new project in the Add New Project form. All fields are mandatory.
 - a. The Project ID must be a whole number. All project IDs must be unique.
 - b. The Title is the project title. Any character is accepted. Foreign languages are supported.
 - c. The Description text field contains the project description. Any Unicode character is supported and foreign languages and writing systems are supported.
 - d. The Assigned Tasks field is where you enter a list of tasks separated by a comma. Unicode characters, foreign languages and writing systems are supported.
4. Click the “Add Project” button.

Add New Project

Project ID: 1

Title: Develop Weather Web Application

Description: Develop a weather forecast and monitoring web application. Include support for different languages including English, French, Spanish, German, 中文, 한국어, 日本語, हिन्दी, తెలుగు, ਪੰਜਾਬੀ, Ελληνικά, Русский, عربی, Українська, سرائی

Assigned Tasks:

- Design Website, Get User Requirements, Generate UML Use Case Diagrams, Generate UML Class Diagrams, Create UML ERD Diagrams, Create Mockups and Wireframes, Code the Web Application Backend, Code the Web Application Frontend, Create and commit to Git repository for version control, Test Application, Build Application, Set up CI/CD pipelines, Deploy Application, Choose development methodology, Create Architecture Diagram, Organise DevOps team.

ADD PROJECT

Figure 142 - Add New Project Form.

- You will be redirected to the View Projects page where you can view the list of projects as a table.

View Projects			
PROJECT ID	TITLE	DESCRIPTION	ASSIGNED TASKS
1	Develop Weather Web Application	Develop a weather forecast and monitoring web application. Include support for different languages including English, French, Spanish, German, 中文, 한국어, 日本語, हिन्दी, తెలుగు, ਪੰਜਾਬੀ, Ελληνικά, Русский, عربی, Українська, سرائی	Design Website, Get User Requirements, Generate UML Use Case Diagrams, Generate UML Class Diagrams, Create UML ERD Diagrams, Create Mockups and Wireframes, Code the Web Application Backend, Code the Web Application Frontend, Create and commit to Git repository for version control, Test Application, Build Application, Set up CI/CD pipelines, Deploy Application, Choose development methodology, Create Architecture Diagram, Organise DevOps team.

Figure 143 - Project added to View Projects Page.

View Projects

- You can view the list of projects as a table by clicking “View Projects” in the “Manage Projects” dropdown menu.

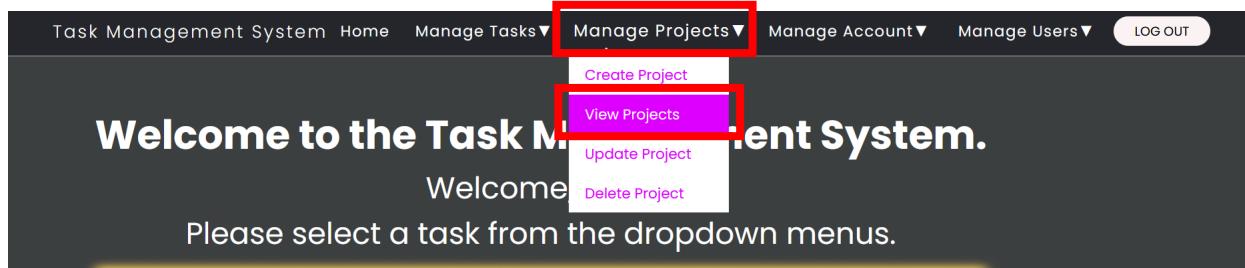


Figure 144 - Select "View Projects" from "Manage Projects" dropdown menu.

2. You will be redirected to the View Projects that displays a table of all the projects. See the two screenshots for examples.

View Projects			
PROJECT ID	TITLE	DESCRIPTION	ASSIGNED TASKS
1	Develop Weather Web Application and Mobile App	UPDATED Develop a weather forecast and monitoring web application. Include support for different languages including English, French, Spanish, German, 中文, 한국어, 日本語, हिन्दी, తెలుగు, தமிழ், ఇంగ్లీష్, ఎల్లాము, రుస్సియింగ్, ఔరియా, ఉక్రైనీస్కి	Design Website, Get User Requirements, Generate UML Use Case Diagrams, Generate UML Class Diagrams, Create UML ERD Diagrams, Create Mockups and Wireframes, Code the Web Application Backend, Code the Web Application Frontend, Create and commit to Git repository for version control, Test Application, Build Application, Set up CI/CD pipelines, Deploy Application, Choose development methodology, Create Architecture Diagram, Organise DevOps team.
2	Create a FPS Video Game	Create a First Person Shooter Video Game using Unreal Engine 5.	Design Game, Design UML diagrams, Create Mockups, Create 3D Models, Create Animations, Create 3D Textures, Create Game using Unreal Engine 5, Code the Game Using Visual Studio and C++, Build the Game, Test the Game, Deploy and Release the Game

Figure 145 - View Projects page.

3. Select "Create Project" from "Manage Projects" dropdown menu.

View Projects			
PROJECT ID	TITLE	DESCRIPTION	ASSIGNED TASKS
1	Create Wrb App	Develop Web Application	Plan Web App, Create UML diagrams, Code Website, Test Website
2	project	project project	project project project project project project project project
3	Update Project 3	Update Project 3Update Project 3Update Project 3	task 1, task 2, task 3
5	project xxxxxxxx	This is project xxxx	t1, t2, t3
6	project	project	project
7	project	project	project
8	project	project	project
10	project	project	project
11	project	project	project
12	project	project	project
13	project	project	project
14	project	project	project
15	project	project	project
16	project	project	project
17	project	project	project
18	project	project	project

Figure 146 - View Projects page with example data.

Update a Project – Administrators and Root User Only

- Select “Update Project” from the “Manage Projects” dropdown menu.

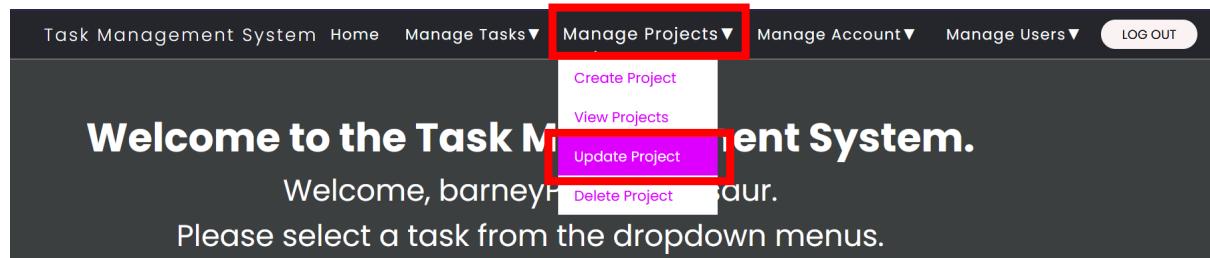


Figure 147 - Select "Update Project" from "Manage Projects" dropdown menu.

- You will be redirected to the Update Project form.
- Fill in the details of the new project in the Add New Project form. All fields are mandatory.
 - The Project ID must be a whole number. All project IDs must be unique.
 - The Title is the project title. Any character is accepted. Foreign languages are supported.
 - The Description text field contains the project description. Any Unicode character is supported and foreign languages and writing systems are supported.
 - The Assigned Tasks field is where you enter a list of tasks separated by a comma. Unicode characters, foreign languages and writing systems are supported.

The screenshot shows the 'Update Project' form. The URL in the browser is 'localhost:5000/updateproject'. The form has the following fields:

- Project ID:** A text input field containing the value '1'.
- Title:** A text input field containing the value 'Develop Weather Web Application and Mobile App'.
- Description:** A text area containing the following text: 'UPDATED Develop a weather forecast and monitoring web application. Include support for different languages including English, French, Spanish, German, 中文, 한국어, 日本語, हिन्दी, తెలుగు, ప్రణామి, Ελληνικά, Русский, اردو, Українська, سوچی'.
- Assigned Tasks:** A text area containing a list of tasks: 'Design Website, Get User Requirements, Generate UML Use Case Diagrams, Generate UML Class Diagrams, Create UML ERD Diagrams, Create Mockups and Wireframes, Code the Web Application Backend, Code the Web Application Frontend, Create and commit to Git repository for version control, Test Application, Build Application, Set up CI/CD pipelines, Deploy Application, Choose development methodology, Create Architecture Diagram, Organise DevOps team.'
- UPDATE PROJECT**: A red button at the bottom of the form.

Figure 148 - Update Project form.

- Click the “Update Project” button.
- You will be redirected to the View Projects page where you can view the updated project.

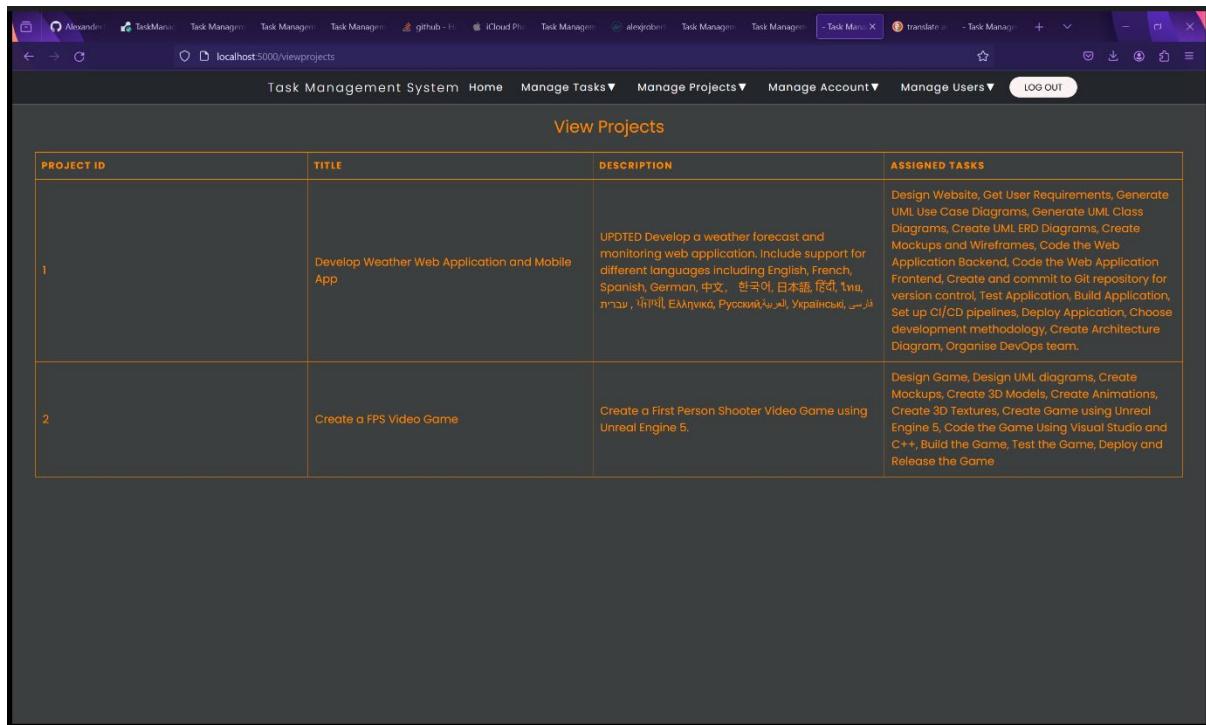


Figure 149 - View Projects page showing updated project.

Delete a Project – Administrators and Root User Only

1. Select “Delete Project” from the “Manage Projects” dropdown menu.

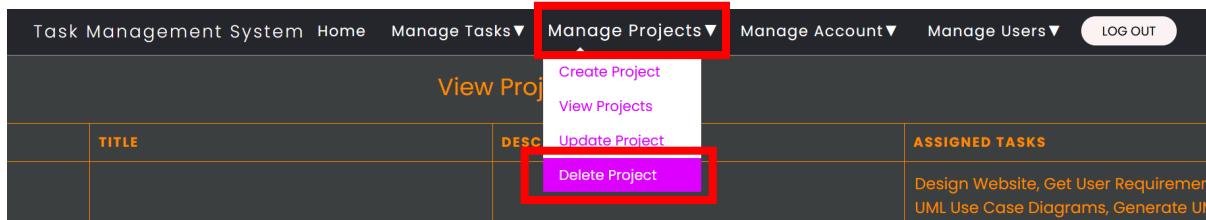


Figure 150 - Select "Delete Project" from "Manage Projects" dropdown menu.

2. You will be redirected to the Delete Project form.
 3. Enter both the Project ID and Project Title into the form. The Project ID and Project Title must match to help prevent accidental deletion. In the example screenshot, Project 2 will be deleted.

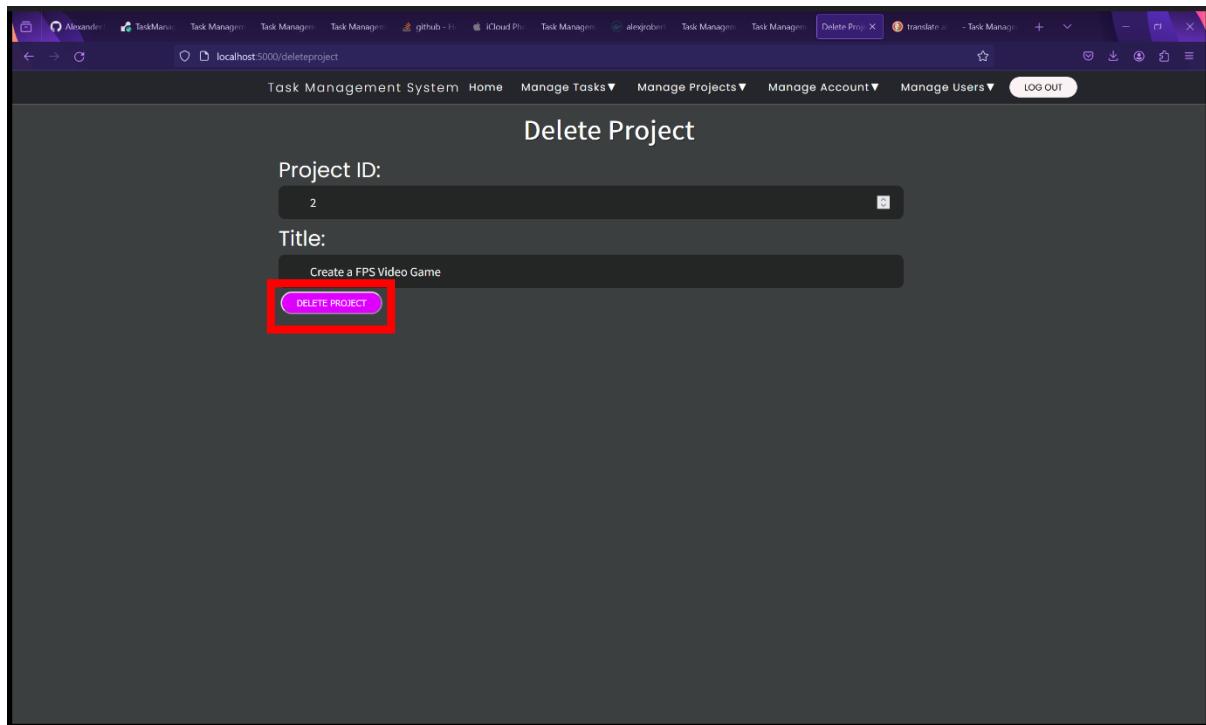


Figure 151 - Delete Project form.

- Click the “Delete Project” button.
- You will be redirected to the View Projects page, and you will see that the project has been deleted from the database.

View Projects			
PROJECT ID	TITLE	DESCRIPTION	ASSIGNED TASKS
1	Develop Weather Web Application and Mobile App	UPDATED Develop a weather forecast and monitoring web application. Include support for different languages including English, French, Spanish, German, 中文, 한국어, 日本語, 泰语, ไทย, ភាសាខ្មែរ, Ελληνικά, Русский, اردو, Українська, سریانی	Design Website, Get User Requirements, Generate UML Use Case Diagrams, Generate UML Class Diagrams, Create UML ERD Diagrams, Create Mockups and Wireframes, Code the Web Application Backend, Code the Web Application Frontend, Create and commit to Git repository for version control, Test Application, Build Application, Set up CI/CD pipelines, Deploy Application, Choose development methodology, Create Architecture Diagram, Organise DevOps team.

Figure 152 - Project deleted from View Projects page.

Manage Tasks

Create a New Task

- Select “Create Task” from the “Manage Tasks” dropdown menu.

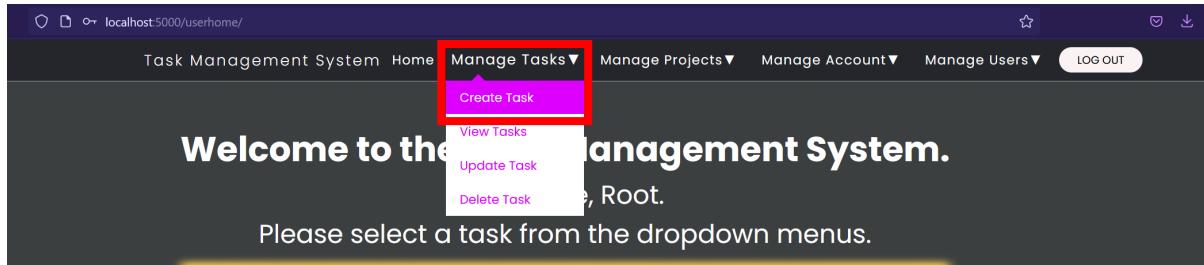


Figure 153 - Select "Create Task" from "Manage Tasks" dropdown menu.

- You will be redirected to the Add New Task form.
- Fill in the details of the new task in the Add New Task form. All fields are mandatory.
 - Task ID must be a whole number and unique.
 - Title is the title of the task. Unicode, foreign languages and writing systems are supported.
 - Description is the description of the task. This text field is resizable and supports Unicode, foreign languages and writing systems.
 - Due Date is the date the project is due. Must be in the format DD/MM/YYYY. Click the calendar icon in the date field for the menu to appear to select a date.

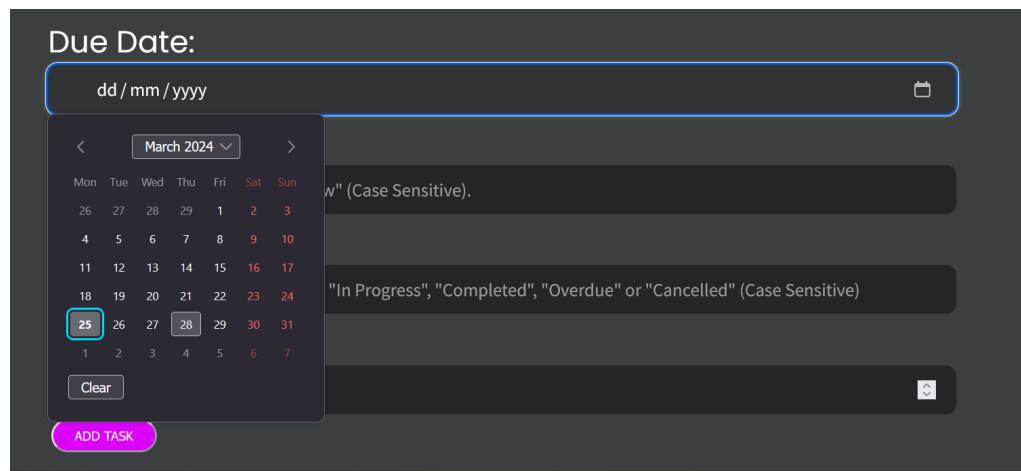


Figure 154 - Callender on Create Task form.

- Priority is how urgently the task must be completed. Priority can only be “High”, “Medium” or “Low” and is case sensitive.
- Status is the status of the task. Status can only be “Not Started”, “Planned”, “In Progress”, “Completed”, “Overdue”, or “Cancelled” and is case sensitive.
- The Project ID must be a whole number and correspond to the Project ID of a project stored in the Task Management System’s database.
- Then click the “Add Task” button.

Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ Manage Users▼ LOG OUT

Add New Task

Task ID:

Title:

Description:

Due Date:

Priority:

Status:

Project ID:

ADD TASK

Figure 155 - Create Task form.

5. You will be redirected to the View Tasks page that displays all the tasks as a table. The task you have added will be available here.

Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ Manage Users▼ LOG OUT

View Tasks

TASK ID	TITLE	DESCRIPTION	DUUE DATE	PRIORITY	STATUS	PROJECT ID
1	Design Website	Design the Weather Forecast website and web app.	2024-03-29	High	In Progress	1

Figure 156 - New task added to View Tasks page.

View Tasks

1. Select “View Tasks” from the “Manage Tasks” dropdown menu.

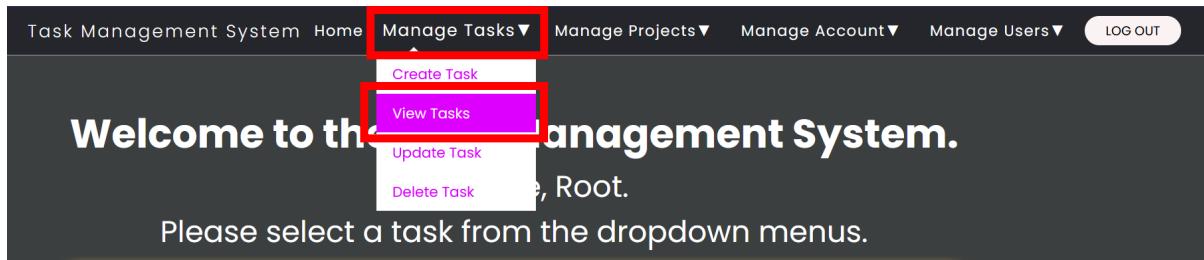


Figure 157 - Select "View Tasks" from "Manage Tasks" dropdown menu.

2. You will be redirected to the View Tasks page that displays a list of tasks and details of each task as a table.

Task ID	Title	Description	Due Date	Priority	Status	Project ID
1	Design Website	Design the Weather Forecast website and web app.	2024-03-29	High	In Progress	1
2	Get User Requirements	Study the user and client requirements	2024-04-01	High	Planned	1
3	Generate UML Use Case Diagrams	Create Use Case Diagram for Weather Web App	2024-04-05	High	In Progress	1
4	Generate UML Class Diagrams	Create Class Diagrams for Users and Weather Web App Functions	2024-04-05	High	In Progress	1
5	Create UML ERD Diagrams	Create Entity Relationship Diagrams for database and data associated with users and application	2024-04-05	High	In Progress	1
6	Create Mockups and Wireframes	Create Wireframe and Mockups of the Weather Web Application and Mobile App.	2024-04-19	Medium	In Progress	1
7	Code the Web Application Backend	Code the backend of the Weather Web Application using Python 3 with Flask Framework and SQL.	2024-05-24	Medium	Planned	1
8	Code the Web Application Frontend	Code the front end of the Weather Web Application using HTML5, CSS and JavaScript.	2024-05-24	Medium	Planned	1
	Create and commit to Git	Create a GitHub repository to				

Figure 158 - View Tasks page with example tasks.

Update Tasks – Administrators and Root User Only

1. Select "Update Task" from the "Manage Tasks" dropdown menu.

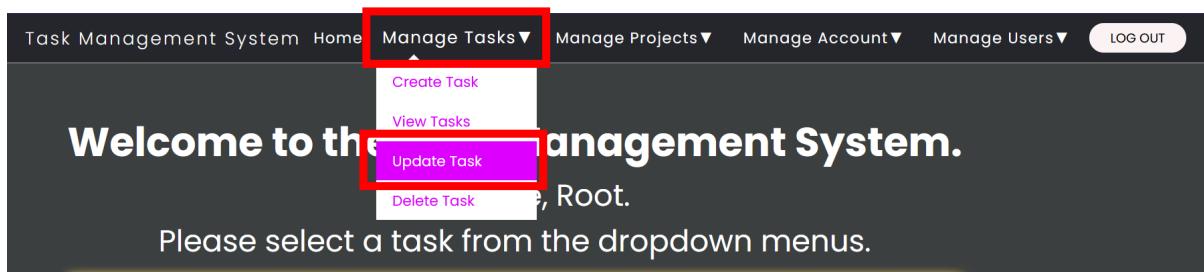


Figure 159 - Select "Update Task" from "Manage Tasks" dropdown menu.

2. Fill in the details of the new task in the Update Task form. All fields are mandatory.
 - a. Task ID must be a whole number and unique.
 - b. Title is the title of the task. Unicode, foreign languages and writing systems are supported.

- c. Description is the description of the task. This text field is resizable and supports Unicode, foreign languages and writing systems.
- d. Due Date is the date the project is due. Must be in the format DD/MM/YYYY. Click the calendar icon in the date field for the menu to appear to select a date.

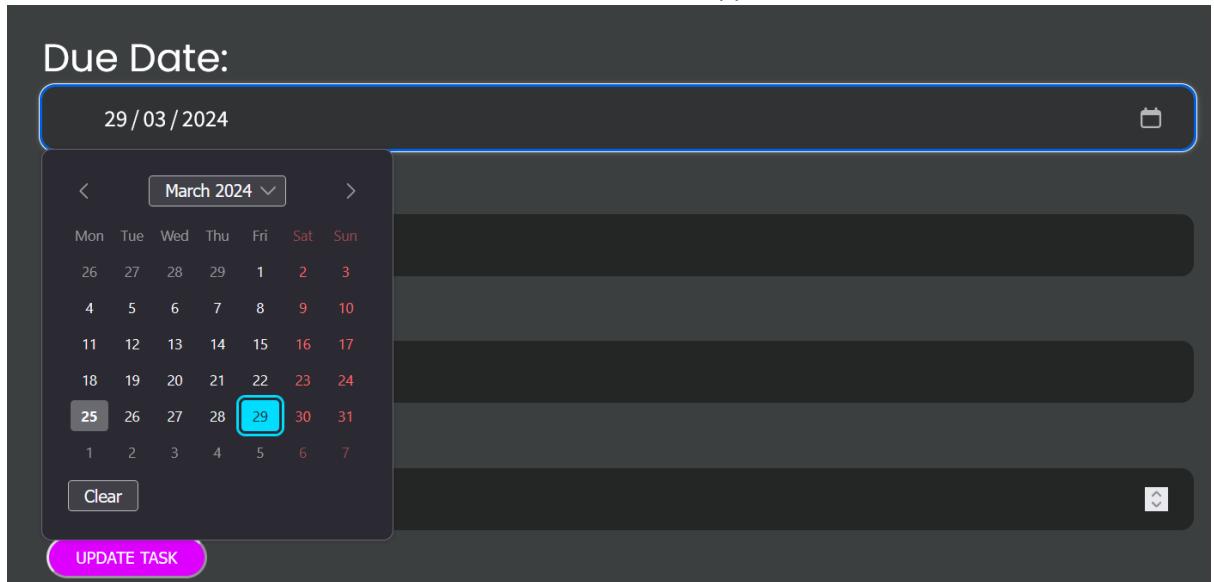


Figure 160 - Callendar on Create Task form.

- e. Priority is how urgently the task must be completed. Priority can only be "High", "Medium" or "Low" and is case sensitive.
 - f. Status is the status of the task. Status can only be "Not Started", "Planned", "In Progress", "Completed", "Overdue", or "Cancelled" and is case sensitive.
 - g. The Project ID must be a whole number and correspond to the Project ID of a project stored in the Task Management System's database.
3. Then click the "Update Task" button.

Figure 161 - Update Tasks form.

4. You will be redirected to the View Tasks page where you can view the updated task.

View Tasks						
Task ID	Title	Description	Due Date	Priority	Status	Project ID
1	Design Website UPDATED	Design the Weather Forecast website and web app. UPDATED	2024-03-29	High	Completed	1
2	Get User Requirements	Study the user and client requirements	2024-04-01	High	Planned	1
3	Generate UML Use Case Diagrams	Create Use Case Diagram for Weather Web App	2024-04-05	High	In Progress	1
4	Generate UML Class Diagrams	Create Class Diagrams for Users and Weather Web App Functions	2024-04-05	High	In Progress	1

Figure 162 - View Tasks page with updated task.

Delete Tasks – Administrators and Root users only.

1. Select “Delete Task” from the “Manage Tasks” dropdown menu.

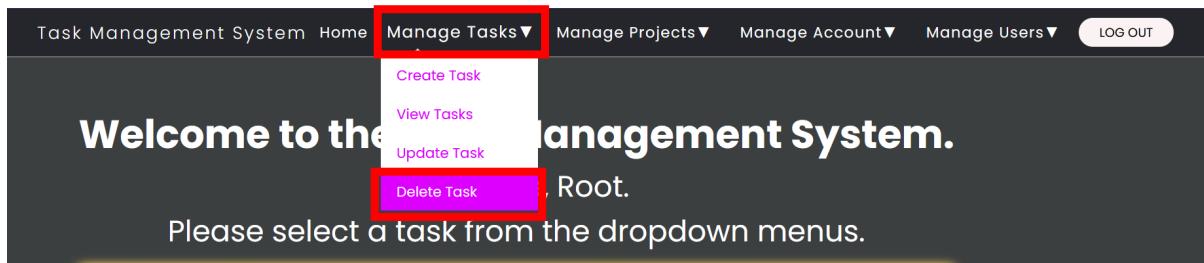
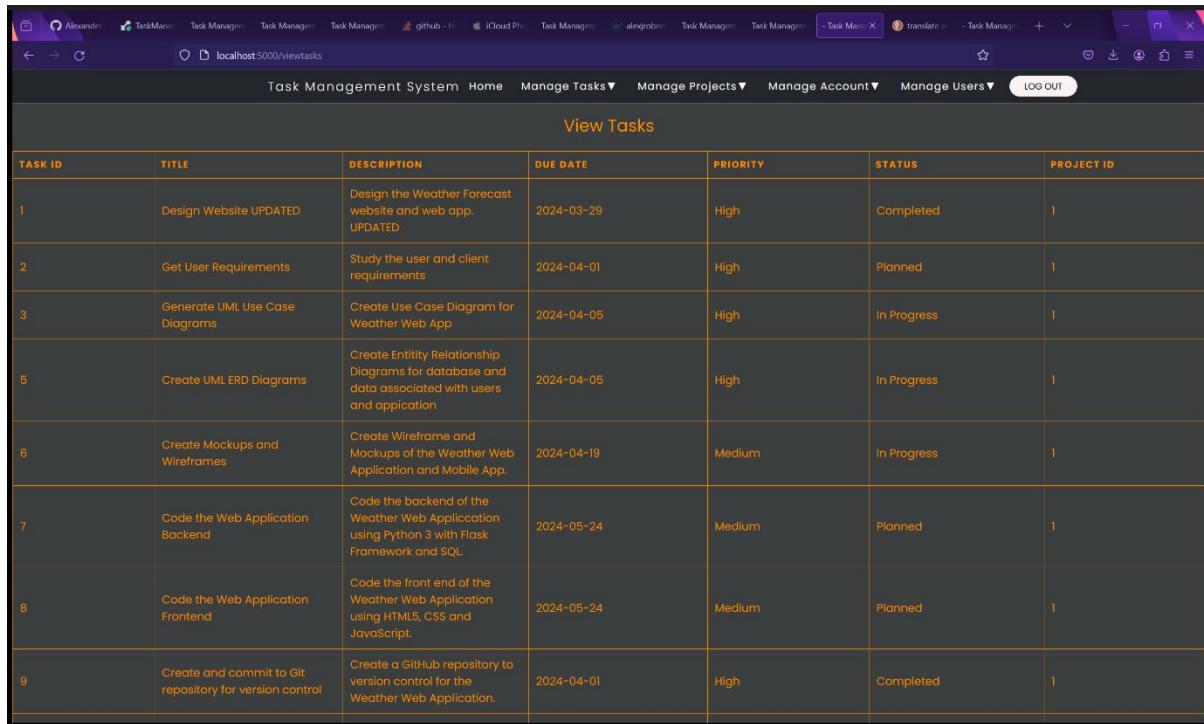


Figure 163 - Select "Delete Task" from "Manage Tasks" dropdown menu.

2. You will be redirected to the Delete Task form.
3. Enter both the Task ID and Task Title into the form. The Task ID and Task Title must match to help prevent accidental deletion. In the example screenshot, Task 4 will be deleted.

Figure 164 - Delete Task form.

4. Click the “Delete Task” button.
5. You will be redirected to the View Tasks page, and you will see that the task (Task 4 in this case) has been deleted from the database.



TASK ID	TITLE	DESCRIPTION	DUUE DATE	PRIORITY	STATUS	PROJECT ID
1	Design Website UPDATED	Design the Weather Forecast website and web app. UPDATED	2024-03-29	High	Completed	1
2	Get User Requirements	Study the user and client requirements	2024-04-01	High	Planned	1
3	Generate UML Use Case Diagrams	Create Use Case Diagram for Weather Web App	2024-04-05	High	In Progress	1
5	Create UML ERD Diagrams	Create Entity Relationship Diagrams for database and data associated with users and application	2024-04-05	High	In Progress	1
6	Create Mockups and Wireframes	Create Wireframe and Mockups of the Weather Web Application and Mobile App.	2024-04-19	Medium	In Progress	1
7	Code the Web Application Backend	Code the backend of the Weather Web Application using Python 3 with Flask Framework and SQL.	2024-05-24	Medium	Planned	1
8	Code the Web Application Frontend	Code the front end of the Weather Web Application using HTML5, CSS and JavaScript.	2024-05-24	Medium	Planned	1
9	Create and commit to Git repository for version control	Create a GitHub repository to version control for the Weather Web Application.	2024-04-01	High	Completed	1

Figure 165 - View Tasks page after Task 4 has been deleted.

Managing Your Account

View Your Account Details

1. Select “View Account Details” from the “Manage Account” dropdown menu.

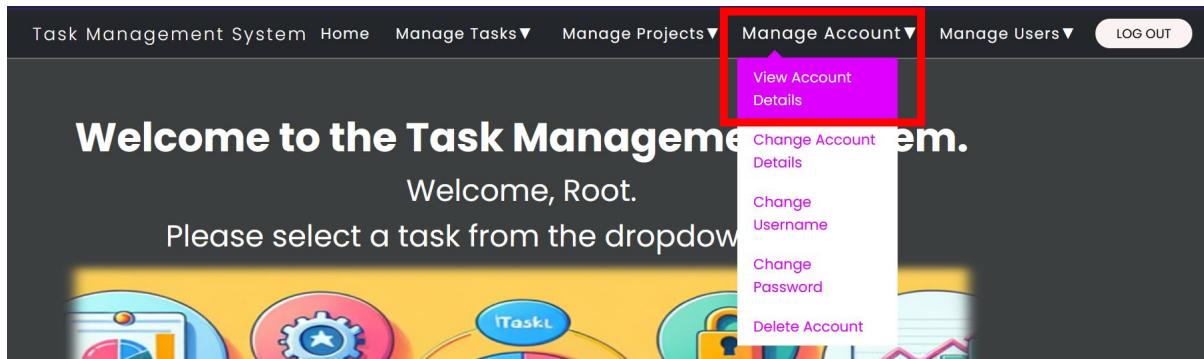


Figure 166 - Select "View Account Details" from "Manage Account" dropdown menu.

2. You will be redirected to the Your Account Details page.

The screenshot shows a web browser window with a dark theme. At the top, there is a navigation bar with various links like 'Task Management System Home', 'Manage Tasks', 'Manage Projects', 'Manage Account', and 'Manage Users'. Below the navigation bar, the title 'Your Account Details' is displayed. A table follows, with columns labeled 'USERNAME', 'FIRST NAME', 'LAST NAME', 'EMAIL ADDRESS', 'PHONE NUMBER', and 'ACCOUNT TYPE'. One row in the table contains the values: Root, John, Smith, s275931@uos.ac.uk, 0487587825, and Administrator.

Figure 167 - Your Account Details page

This screenshot is identical to Figure 167, showing the 'Your Account Details' page with the same table and account information.

Figure 58 - Your Account Details.

Changing your Account Details

1. Select “Change Account Details” from the “Manage Account” dropdown menu.

The screenshot shows a dropdown menu for 'Manage Account' with several options: 'View Account Details', 'Change Account Details' (which is highlighted with a pink rectangle), 'Change Username', 'Change Password', and 'Delete Account'. The background shows a 'Welcome to the Task Management System' message and some decorative icons.

Figure 168 - Select "Change Account Details" from "Manage Account" dropdown menu.

2. You will be redirected to the Update Account Details page where you can change all your account details except your username and password which are done on a different page. You also cannot change your account type or blocked status – only an administrator or Root user can do this.

Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ Manage Users▼ LOG OUT

Update Account Details

Username: finnshark001

Update First Name: Finn-Updated

Update Last Name: Sharkey-Updated

Update Email Address: robertsona97@icloud.com

Update Phone Number: 046263482879

Password: [REDACTED]

UPDATE ACCOUNT DETAILS

Figure 169 - Update Account Details form.

3. Fill in the Update Account Details Form and click the “Update Account Details” button.
4. You will be redirected to the User Home page with a message saying, “Account Details Updated Successfully”.

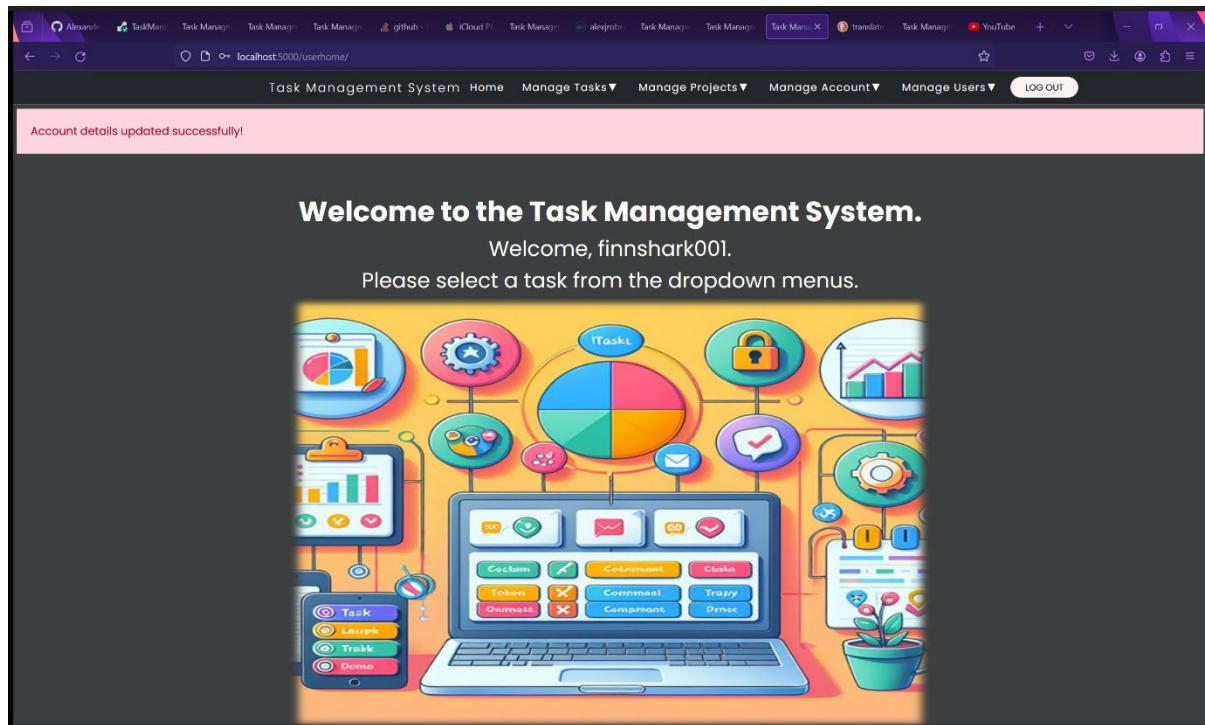


Figure 170 - User Home page after updating account details.

5. You can view your updated account details in the View Account Details page.

Your Account Details					
Username	First Name	Last Name	Email Address	Phone Number	Account Type
finnshark001	Finn-Updated	Sharkey-Updated	robertson97@icloud.com	046263482879	Standard

Figure 171 - Account details after being updated.

Changing your Username

- Select “Change Username” from the “Manage Account” dropdown menu.

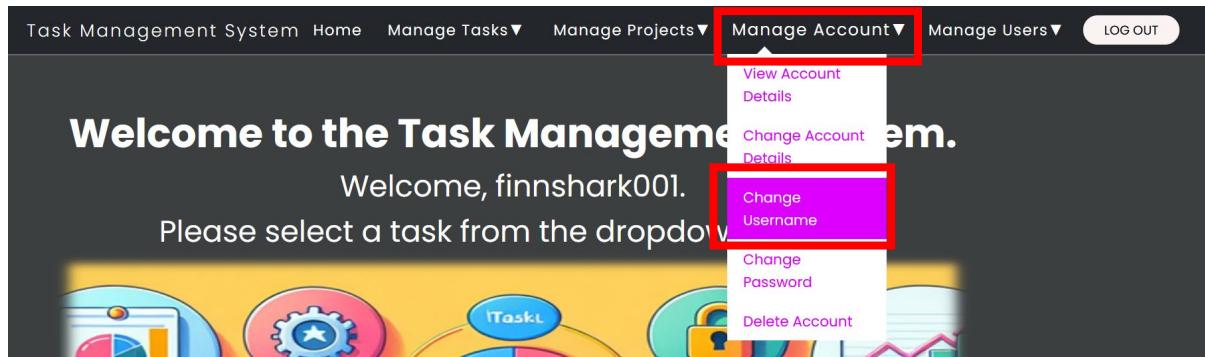


Figure 172 - Select "Change Username" from "Manage Account" dropdown menu.

- You will be required to Change Username form page.
- Fill in the Change Username form with:
 - Your current username
 - Your new username
 - Confirm your new username so you remember it and prevent accidental changes.
 - Enter your password to confirm your username change, for security to prevent unauthorized username changes and to prevent accidental username changes.
- Click the “Change Username” button.

The screenshot shows the 'Change Username' form. It has four input fields: 'Existing Username' (finnshark001), 'New Username' (SharkeyFinn002), 'Confirm New Username' (SharkeyFinn002), and 'Password' (represented by a series of dots). The 'CHANGE USERNAME' button at the bottom is highlighted with a red box.

Figure 173 - Change Username form.

5. You will be redirected to the User Home Page with a message saying, “Username changes successfully!”.

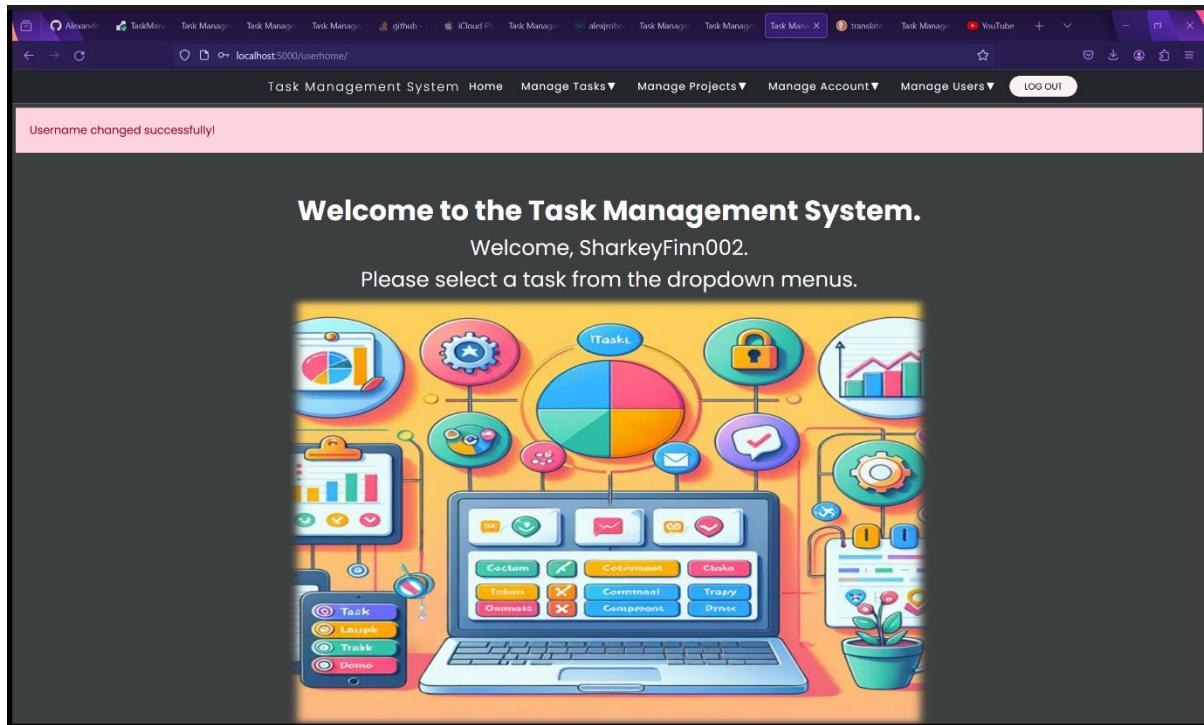


Figure 174 - User Home page after changing username.

6. You can view your username change in the View Account details page.

Your Account Details					
USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE
SharkeyFinn002	Finn-Updated	Sharkey-Updated	robertson97@icloud.com	046263482879	Standard

Figure 175 - Your Account Details page after changing username.

Changing your Password

1. Select “Change Password” from the “Manage Account” dropdown menu.

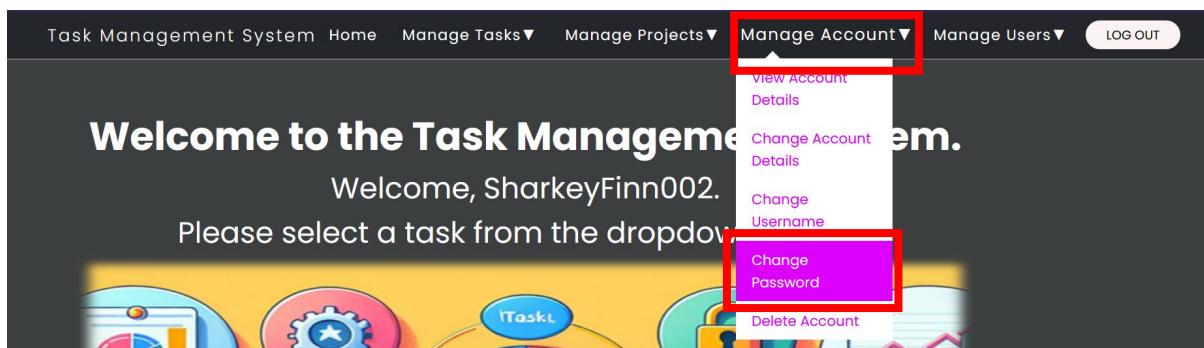


Figure 176 - Select "Change Password" from "Manage Account" dropdown menu.

2. You will be redirected to the Change Password form page.
3. Fill in the Change Password form:

- a. Username – Your current username.
 - b. Existing Password – for security to prevent unauthorised and accidental password changes.
 - c. New Password – must meet the following requirements: Minimum length of 10 characters, at least one capital letter, one lowercase letter, one number, one special ASCII, UTF or Unicode character and must not be a common password.
 - d. Confirm new password so you are less likely to forget it.
4. Click the “Change Password” button.

The screenshot shows a web browser window with multiple tabs open, including 'AlexanderJohnRobert', 'Task Management System', and several 'Change User Password' tabs. The main content area is titled 'Change Password'. It contains four input fields: 'Username' (SharkeyFinn002), 'Existing Password' (redacted), 'New Password' (redacted), and 'Confirm New Password' (redacted). Below these fields is a purple button labeled 'CHANGE PASSWORD', which is highlighted with a red rectangular border.

Figure 177 - Change Password form page.

- a. You will be redirected to the User Home page showing a message saying, “Password Changed Successfully”.

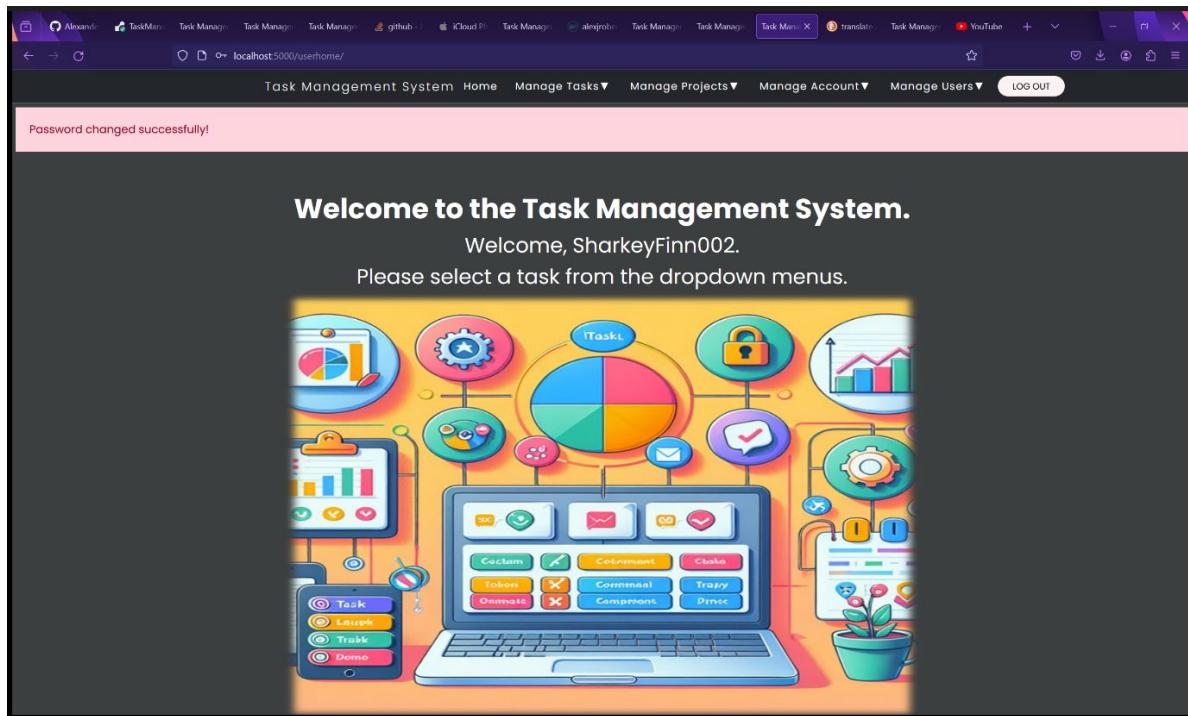


Figure 178 - User Home page after changing password.

5. You will not be able to view your password in the View Account Details page because of security and privacy reasons.

Deleting your Account

WARNING: DELETING YOUR ACCOUNT WILL BE PERMANENT.

1. Select “Delete Account” from the “Manage Account” dropdown menu.

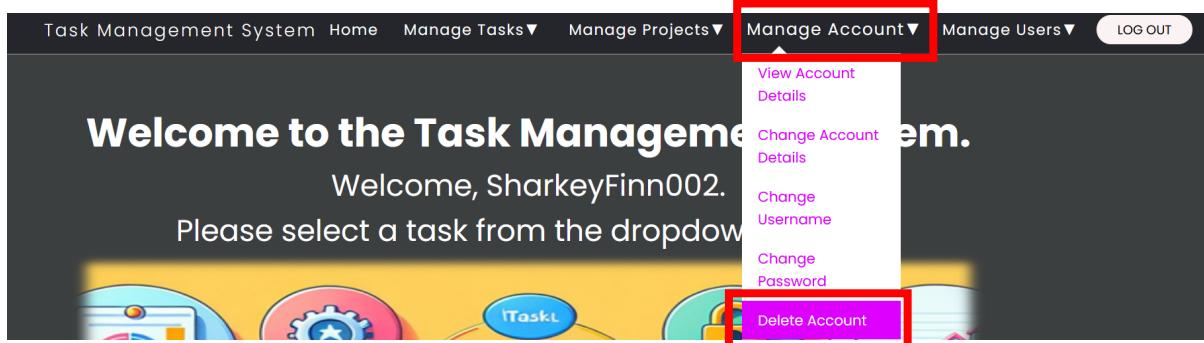


Figure 179 - Select "Delete Account" from "Manage Account" dropdown menu.

2. You will be redirected to the Delete Account form page.
3. Fill in the Delete Account form:
 - a. Username
 - b. Password – security to prevent unauthorized and malicious account deletion.
 - c. Confirm Password – security to prevent accidental account deletion.
4. Click the “Delete Account” button.

A screenshot of a web browser showing the 'Delete Account' page of a Task Management System. The page has a dark background. At the top, there are tabs for 'Task Management System - Home', 'Manage Tasks', 'Manage Projects', 'Manage Account', and 'Manage Users'. A 'LOG OUT!' button is also visible. Below the tabs, the title 'Delete Account' is centered, followed by a red warning message: 'WARNING: This will be permanent.' There are three input fields: 'Username:' containing 'SharkeyFinn002', 'Password:' containing '*****', and 'Confirm Password:' also containing '*****'. A blue button labeled 'DELETE ACCOUNT' is at the bottom, with a red rectangle highlighting it.

Figure 180 - Delete Account form.

5. You will be redirected to the Task Management System landing page with a message saying, "Account Deleted Successfully!".

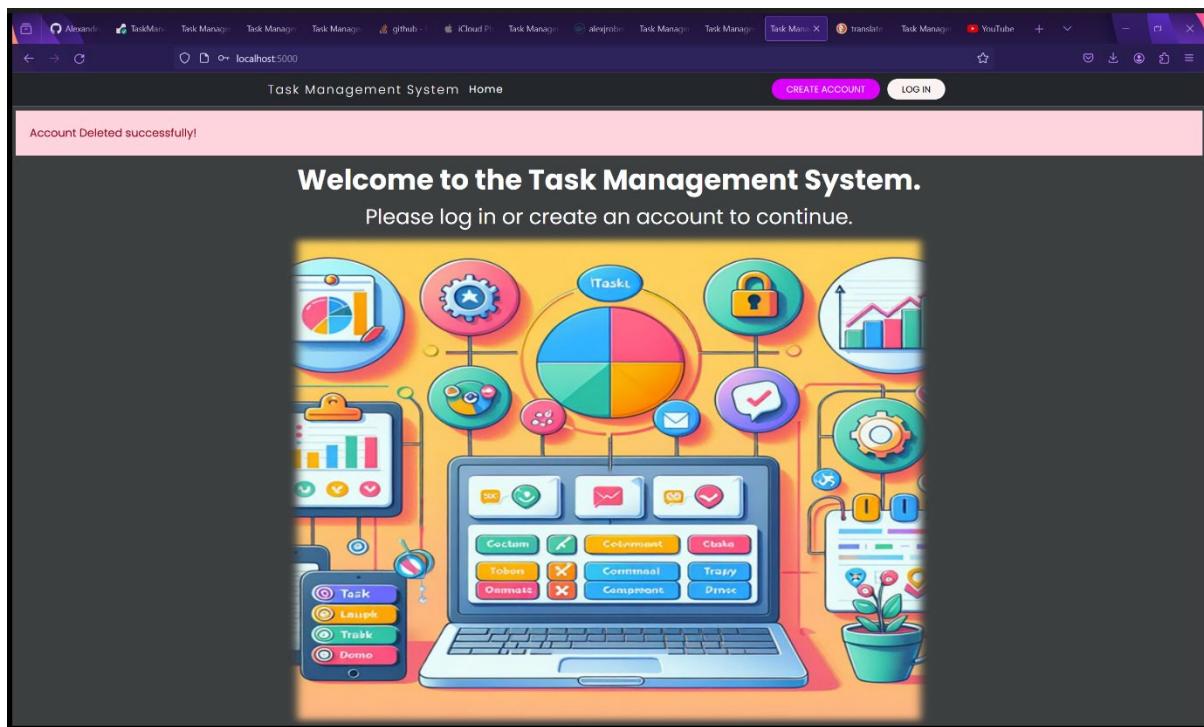


Figure 181 - Landing page after account deletion.

6. You will now be unable to log in proving that your account has been deleted. Instead, you will receive an error message saying, "Username does not exist" (See second screenshot).

S275931

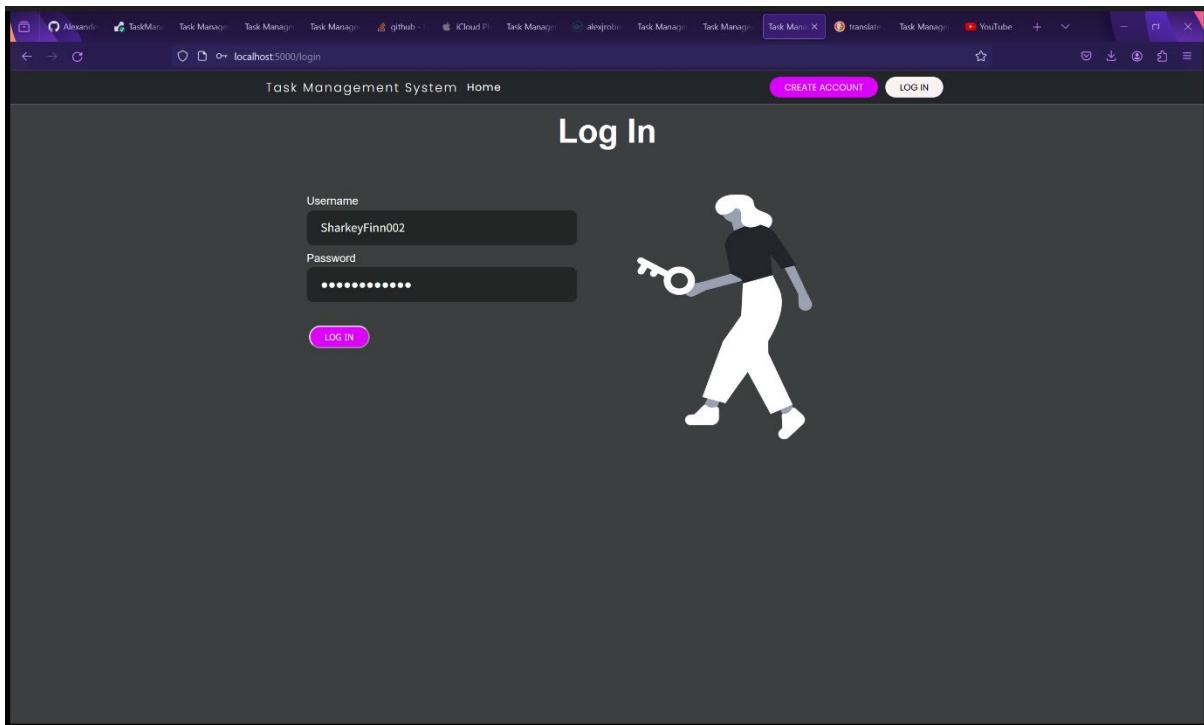


Figure 182 - Log In form.

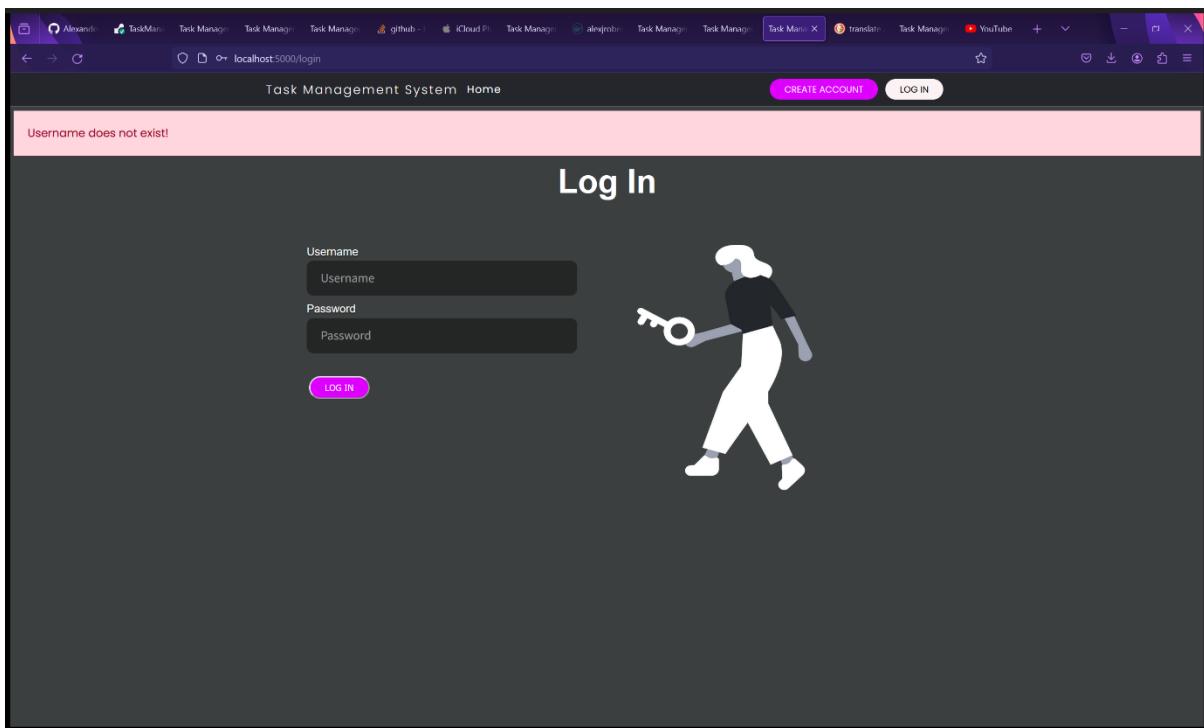


Figure 183 - The user is unable to log in because they have deleted their account.

Managing Users and Administrative Tasks – Administrators and Root User Only

Please Note: Only administrators and Root Users have access to these features. Standard users will be denied access and will receive an Access Denied error.

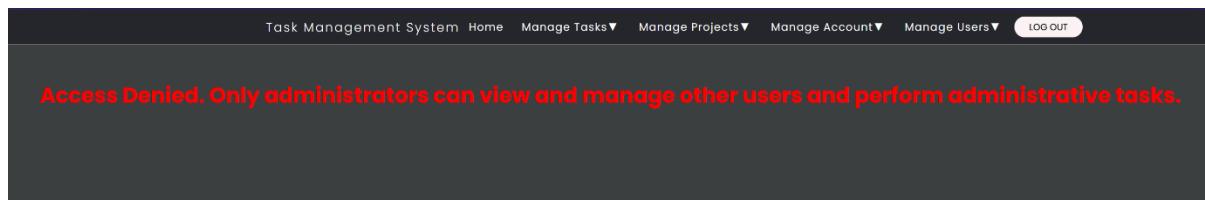


Figure 184 - Error messages that standard users get if they attempt administrative tasks or managing other users.

View the List of Users as a Table (Administrators and Root User Only)

1. Select “View Users” from the “Manage Users” dropdown menu.



Figure 185 - Select "View Users" from "Manage Users" dropdown menu.

2. You will be redirected to the View Users page that displays a list of Users as a table with their details.

The screenshot shows the 'View Users' page. The title 'View Users' is at the top. Below it is a table with the following data:

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Root	John	Smith	s275931@uos.ac.uk	0487587825	Administrator	No
adammccann12	Adam	McCann	robertson97@hotmail.co.uk	07775385490	Standard	No

Figure 186 - View Users page with a list of users and their details.

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Manchester000	Kane	Fenlon	kfenlon@hotmail.com	42342342	Standard	No
LFenlonVolcano	Lauren	Durury	ldurury@msn.com	234223424	Standard	No
GemmaWheels	Gemma	Bird	gbird@hotmail.com	2323466	Standard	No
kellyG	Kelly	Gates	kgates@outlook.com	78624868243	Standard	Yes
ethomas123	Ellie	Thomas	ethomas@icloud.com	836446834	Standard	No
ArasakaCEO123	Bill	Thomas	bthomas@hotmail.com	012434376876	Standard	No
DandyLion000	Daisy	Robinson	drobinson@icloud.com	9835973538	Administrator	No
elasticbands123	Sophie	Rivett	srivett@outlook.com	387548735	Standard	No
FlowerMeadow111	Daisy	Yau	dyau@icloud.com	3546653476	Standard	No
FrankRivett3000	Frank	Rivett	frivett@hotmail.com	93875937457	Standard	No
agillick999	Alex	Gillick	igillick@outlook.com	843428749287	Administrator	No
Birmingham888	Linda	Bone	lbone97@hotmail.co.uk	0392743979	Standard	No
Root	Alexandra	Robinson	s275931@uos.ac.uk	30903853809	Administrator	No
barneyPurpleDinosaur	Alice	Robinson	robertson97@icloud.com	983749739378	Administrator	No

Figure 187 - View Users page with a list of users and their details (example users).

Change User Account Type (Administrators and Root Users Only)

Instructions for changing user account types between Administrator and Standard.

1. Select “Change User Type” from the “Manage Users” dropdown menu.



Figure 188 - Select "View Users" from "Manage Users" dropdown menu.

2. You will be redirected to the Change User Account Type form page.
3. Fill in the Change User Account Type form:
 - a. Username – the username of the account to be changed.
 - b. User Account Type – Type “Administrator” or “Standard” (case sensitive).
4. Click the “Change User Account Type” button.

Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ Manage Users▼ LOG OUT

Change User Account Type

Username:
adammccann12.

User Account Type:
Administrator

CHANGE USER ACCOUNT TYPE

Figure 189 - Change User Account Type form.

5. You will be redirected to the View Users page showing the updated user type. A message will be displayed saying “Account type changed successfully”.

Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ Manage Users▼ LOG OUT

Account type changed successfully!

View Users

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Root	John	Smith	s275931@uos.ac.uk	0487587825	Administrator	No
adammccann12	Adam	McConn	robertson97@hotmail.co.uk	07775385490	Administrator	No

Figure 190 - View Users page after changing a user's Account Type.

Delete a User (Administrators and Root Users Only).

WARNING: DELETING A USER WILL BE PERMANENT.

1. Select “Delete User” from the “Manage Users” dropdown menu.



Figure 191 - Select "Delete User" from "Manage Users" dropdown menu.

2. You will be redirected to the Change User Account Type form page.
3. Fill in the Change User Account Type form:
 - a. Username – the username of the account to be changed.
4. Click the “Delete User” button.

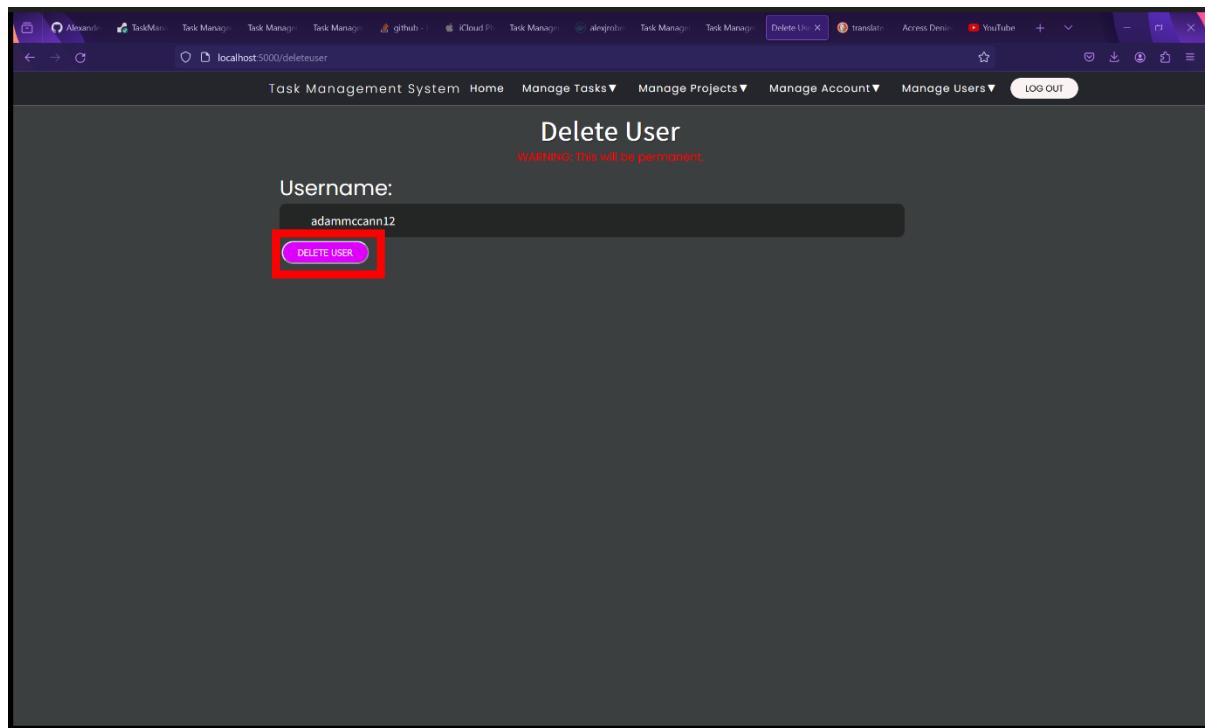


Figure 192 - Delete User form.

5. You will be redirected to the View Users page. A message will be displayed saying “User deleted successfully”.

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Root	John	Smith	s275931@uos.ac.uk	0487587825	Administrator	No

Figure 193 - View Users page after deleting a user.

Blocking a User (Administrators and Root Users Only)

1. Select “Block User” from the “Manage Users” dropdown menu.

- Task Management System Home Manage Tasks▼ Manage Projects▼ Manage Account▼ **Manage Users▼** LOG OUT
- [View Users](#)
- [Change User Type](#)
- [Delete User](#)
- Block User**
- [Unblock User](#)
- [Change User Password](#)
- [Reset Task Management System](#)

Figure 194 - Select "Block User" from "Manage Users" dropdown menu.

2. You will be redirected to the Block User form page.
3. Fill in the Block User form:
 - a. Username – the username of the account to be blocked.
4. Click the “Block User” button.

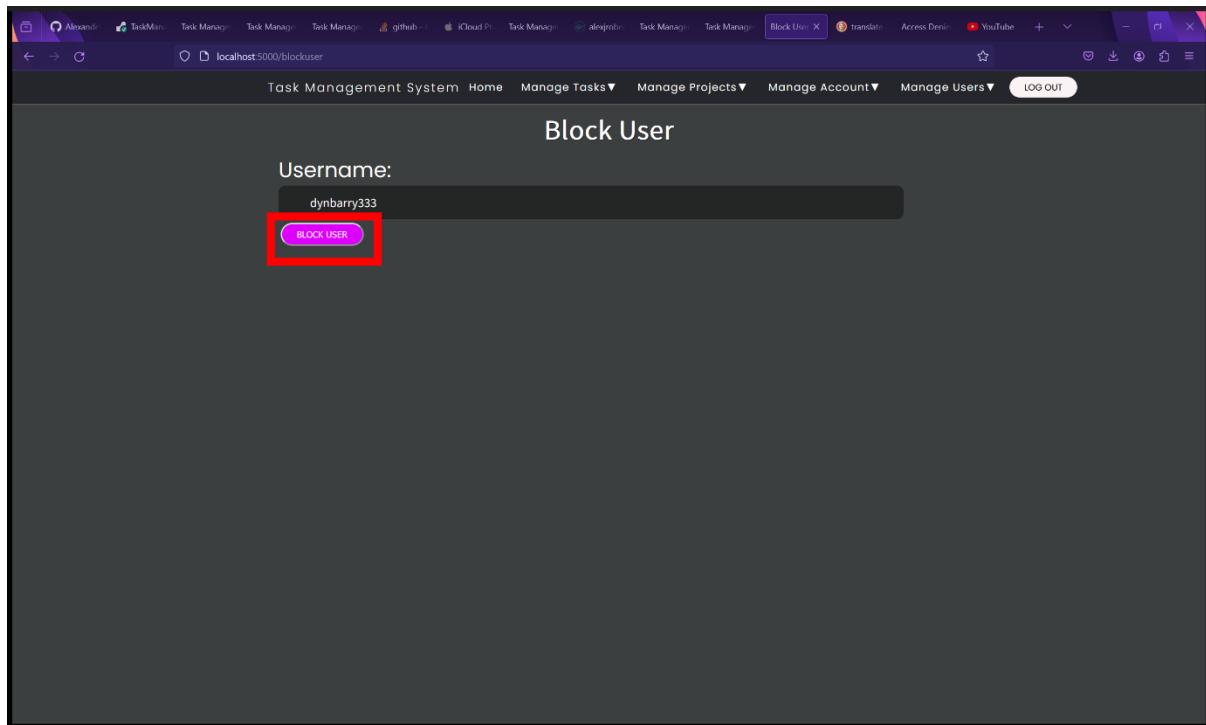


Figure 195 - Block User form.

5. You will be redirected to the View Users page. A message will be displayed saying “User blocked successfully!”. The user’s blocked status will be changed to “Yes”.

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Root	John	Smith	s275931@uos.ac.uk	0487587825	Administrator	No
dynbarry333	Dynosty	Barry	robertson97@icloud.com	072378648	Standard	Yes

Figure 196 - View Users page after blocking a user.

6. The blocked user will now be unable to log in. Instead, they will receive an error message “You have been blocked. Please contact the administrator.”.

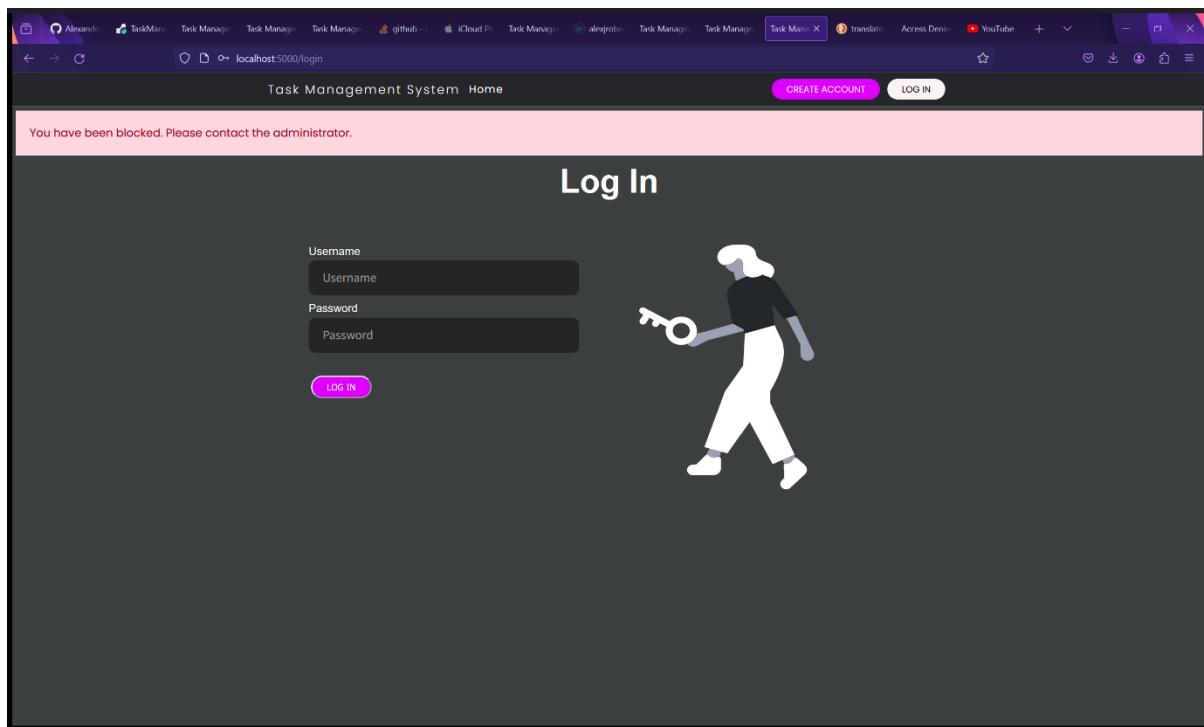


Figure 197 - The blocked user is unable to log in.

Unblocking a User (Administrators and Root Users Only)

1. Select “Unblock User” from the “Manage Users” dropdown menu.



Figure 198 - Select "Unblock User" from "Manage Users" dropdown menu.

2. You will be redirected to the Unblock User form page.
3. Fill in the Unblock User form:
 - a. Username – the username of the account to be unblocked. In the example, we will unblock the same user that we blocked in the previous step.
4. Click the “Unblock User” button.

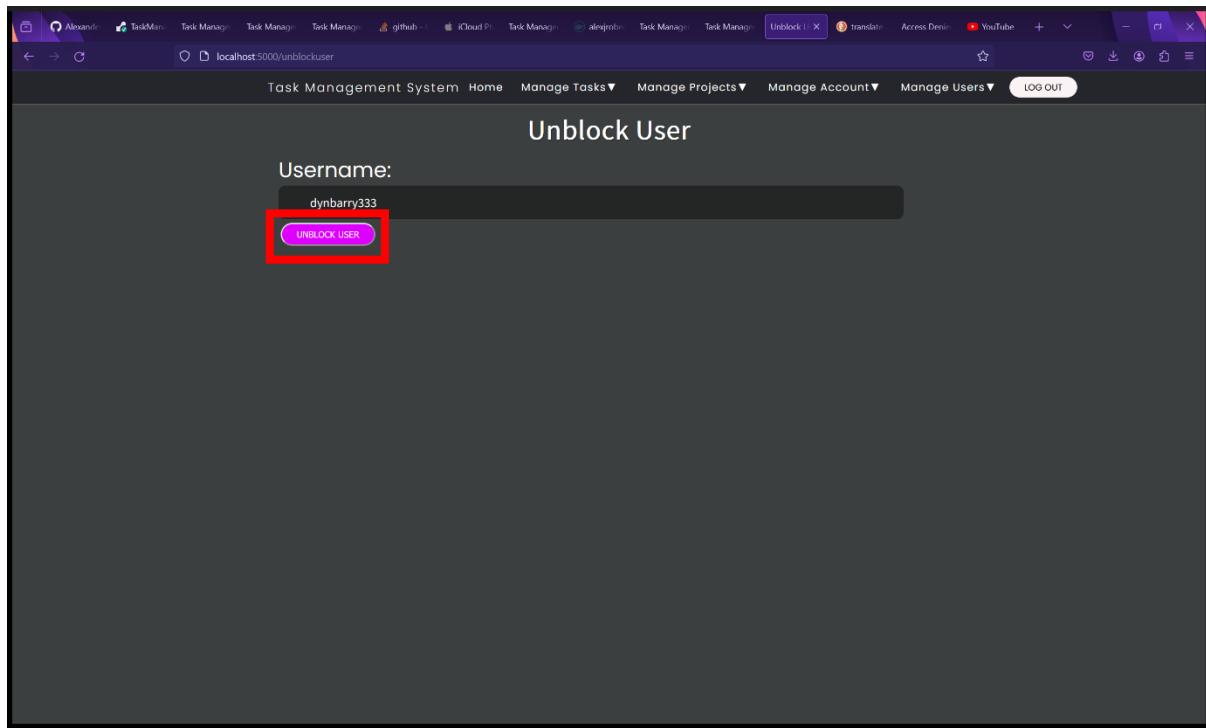


Figure 199 - Unblock User form.

5. You will be redirected to the View Users page. A message will be displayed saying “User unblocked successfully!”. The user’s blocked status will be changed to “No”.

USERNAME	FIRST NAME	LAST NAME	EMAIL ADDRESS	PHONE NUMBER	ACCOUNT TYPE	BLOCKED
Root	John	Smith	s275931@uos.ac.uk	0487587825	Administrator	No
dynbarry333	Dynosty	Barry	robertson97@icloud.com	072378648	Standard	No

Figure 200 - View Users page after unblocking a user.

6. The unblocked user can now log in successfully.

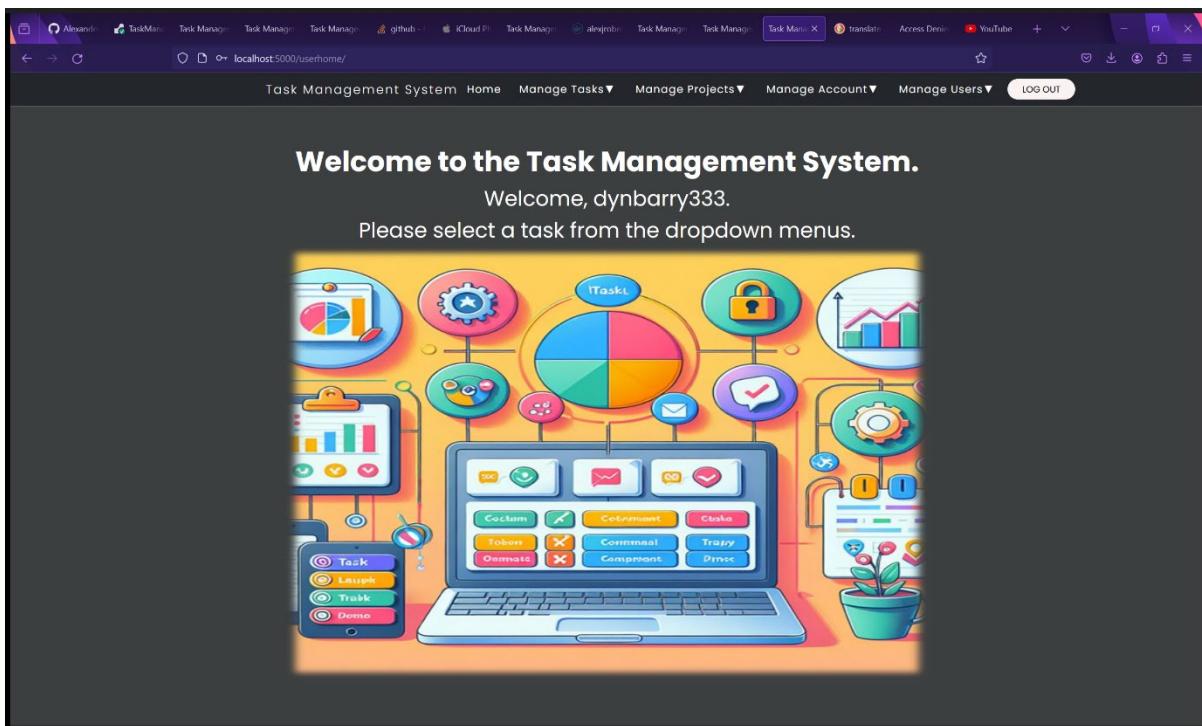


Figure 201 - The user is now able to log in again.

Changing Another User's Password (Administrators and Root Users Only)

1. Select "Change User Password" from the "Manage Users" dropdown menu.



Figure 202 - Select "Change User Password" from "Manage Users" dropdown menu.

2. You will be redirected to the Change User Password form page.
3. Fill in the Change User Password form:
 - a. Username – the username of the account to have their password changed.
 - b. New Password – Enter the new password here. Password must meet the following requirements: Minimum length of 10 characters, at least one uppercase letter, one lowercase letter, one number and one special character. Special characters must be ASCII characters as UTF and Unicode characters are not supported in emailing the new passwords to users and will result in an error. The password must not be a common password.

- Click the “Change User Password” button.

The screenshot shows a 'Change User Password' form. It includes fields for 'Username' (containing 'dynbarry333'), 'New Password', and 'Confirm New Password'. Below these fields is a prominent purple button labeled 'CHANGE USER PASSWORD', which is highlighted with a red rectangular border.

Figure 203 - Change User Password form.

- You will be redirected to the View Users page. A message will be displayed saying “Password Changed successfully!”.
- An email will automatically be sent to the user’s email address.

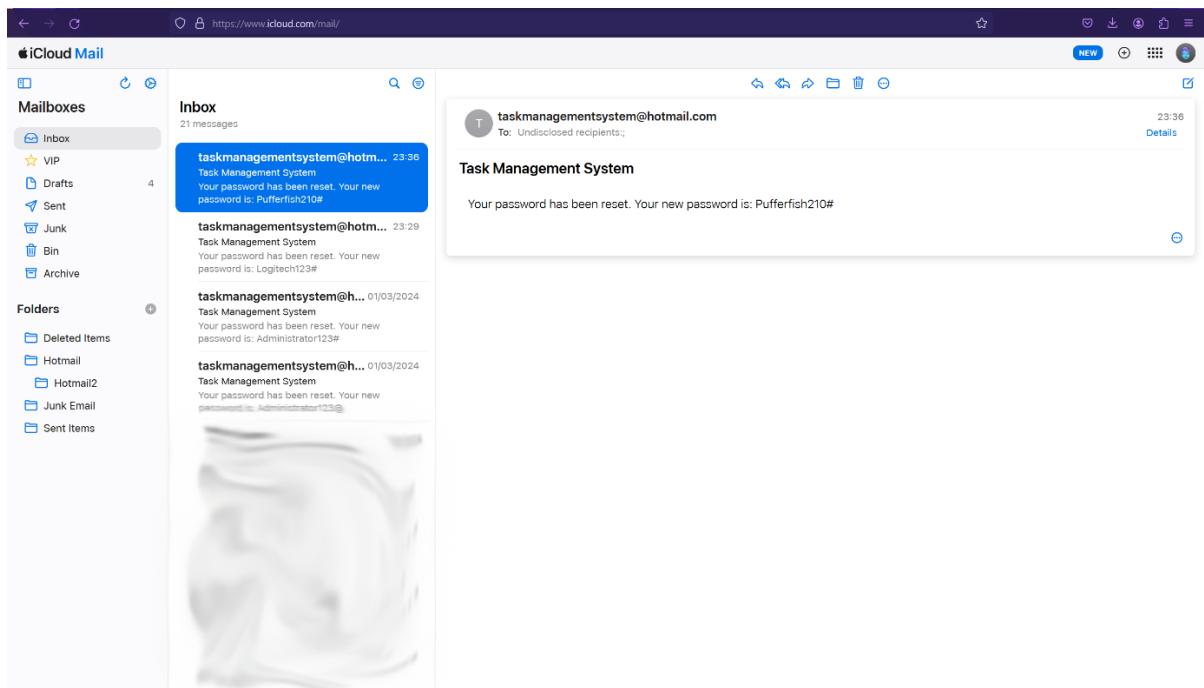


Figure 204 - Automated email sent to user of who’s password has been changed.

Factory Resetting the Task Management System (Administrators and Root Users Only)

**WARNING: THIS WILL BE PERMANENT. ALL DATA WILL BE LOST. IF YOU WANT TO KEEP THE DATA
MAKE A BACKUP FIRST.**

1. Select “Reset Task Management System” from the “Manage Users” dropdown menu.



Figure 205 - Select "Reset Task Management System" from "Manage Users" dropdown menu.

2. You will be redirected to the Reset Task Management System form page.
3. Fill in the Reset Task Management System form:
 - a. Administrator Password – Must be the password of your administrator or Root account – A security measure to prevent unauthorized and malicious factory resets resulting in unauthorized data loss.
 - b. Confirm Password – Re-enter your password. Security measure to prevent accidental factory resets resulting in accidental data loss.
4. Click the “Reset Task Management System” button.

The screenshot shows the 'Reset Task Management System' form. The title is 'Reset Task Management System' with a warning message 'WARNING: This will be permanent.' Below the title are two input fields: 'Administrator Password:' and 'Confirm Password:', both containing redacted text. At the bottom of the form is a blue button labeled 'RESET TASK MANAGEMENT SYSTEM' which is highlighted with a red box.

Figure 206 - Reset Task Management form.

5. The Task Management System will be factory reset and all data (projects, tasks, and users) will be wiped. You will be redirected to the Setup page. A message will appear saying “Task Management System Reset Successfully!” above the setup form.

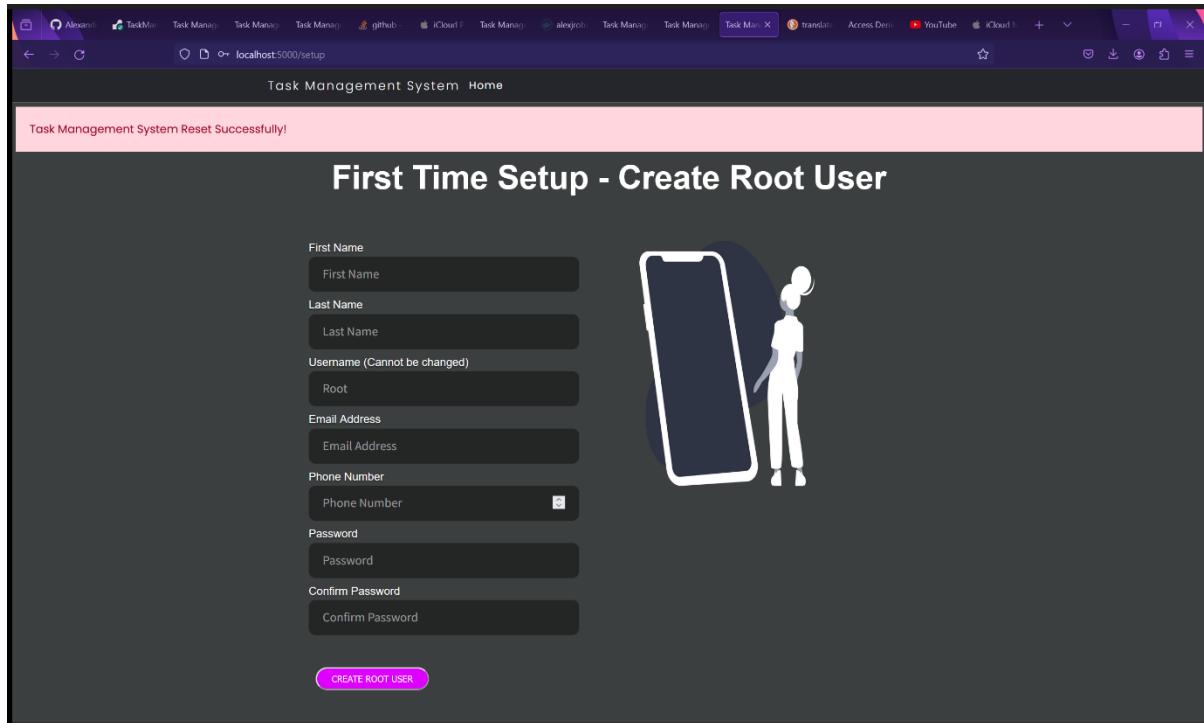


Figure 207 - Setup page after factory resetting the Task Management System.

6. Fill in the Setup form (see First Time Setup) for full instructions.

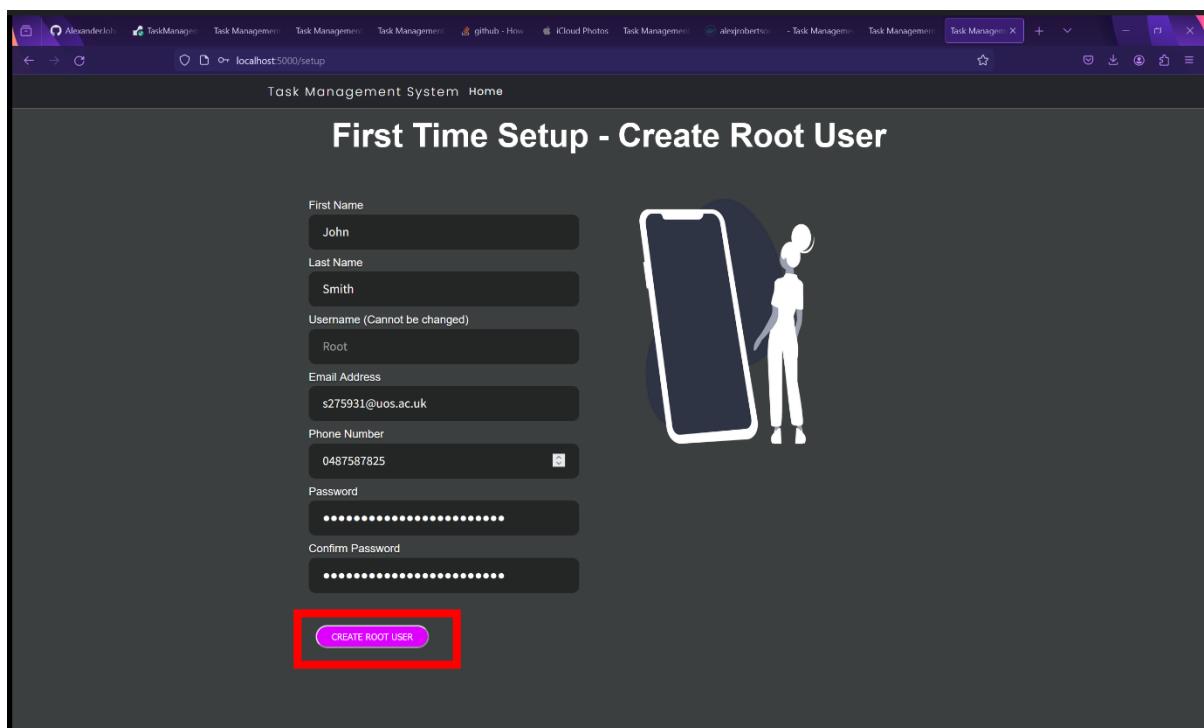


Figure 208 - Completed Setup form.

7. The Task Management System has been reset and is ready for use again.

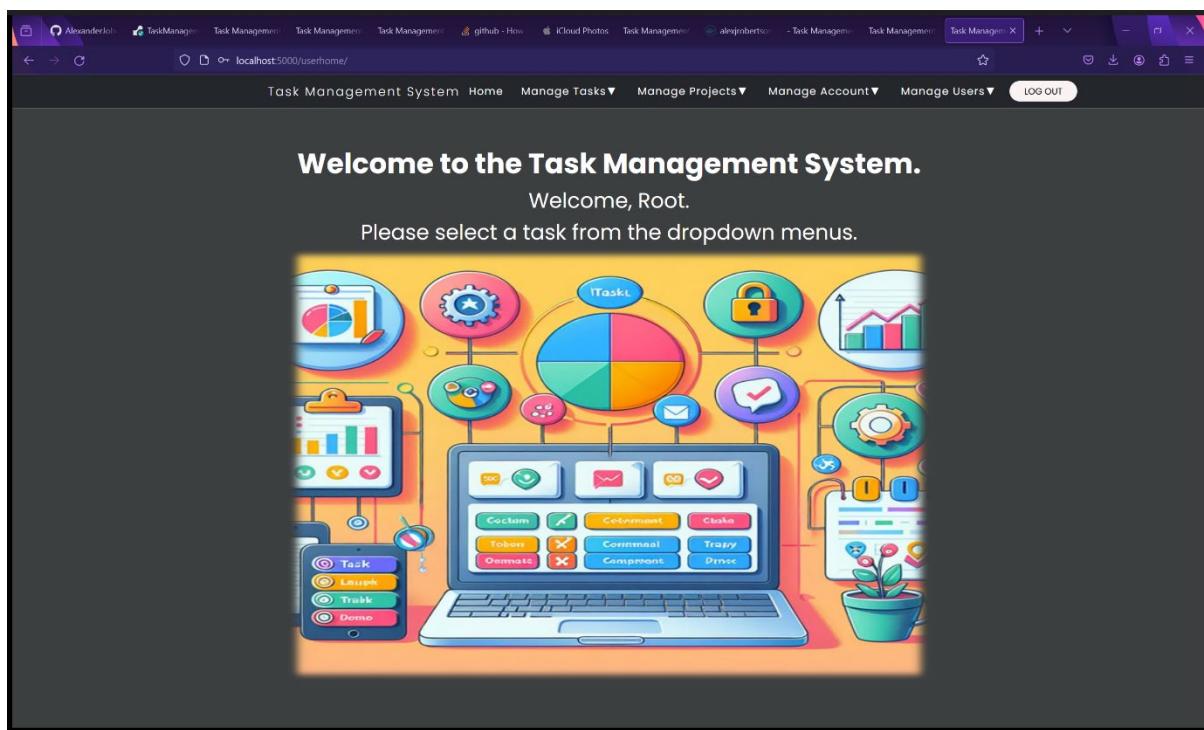


Figure 209 - The Task Management System has been reset.

Task Management System on Mobile Devices.

The Task Management System is a responsive web application and will therefore be displayed differently on mobile devices and displays that are too small or the resolution is too low to display the full navbar or tables. On these devices, the dropdown menus in the navbar can be accessed through a hamburger menu in the top right corner of the webpages. Each task, project and user are displayed in individual card like tables instead of the full table on mobile devices and small displays.

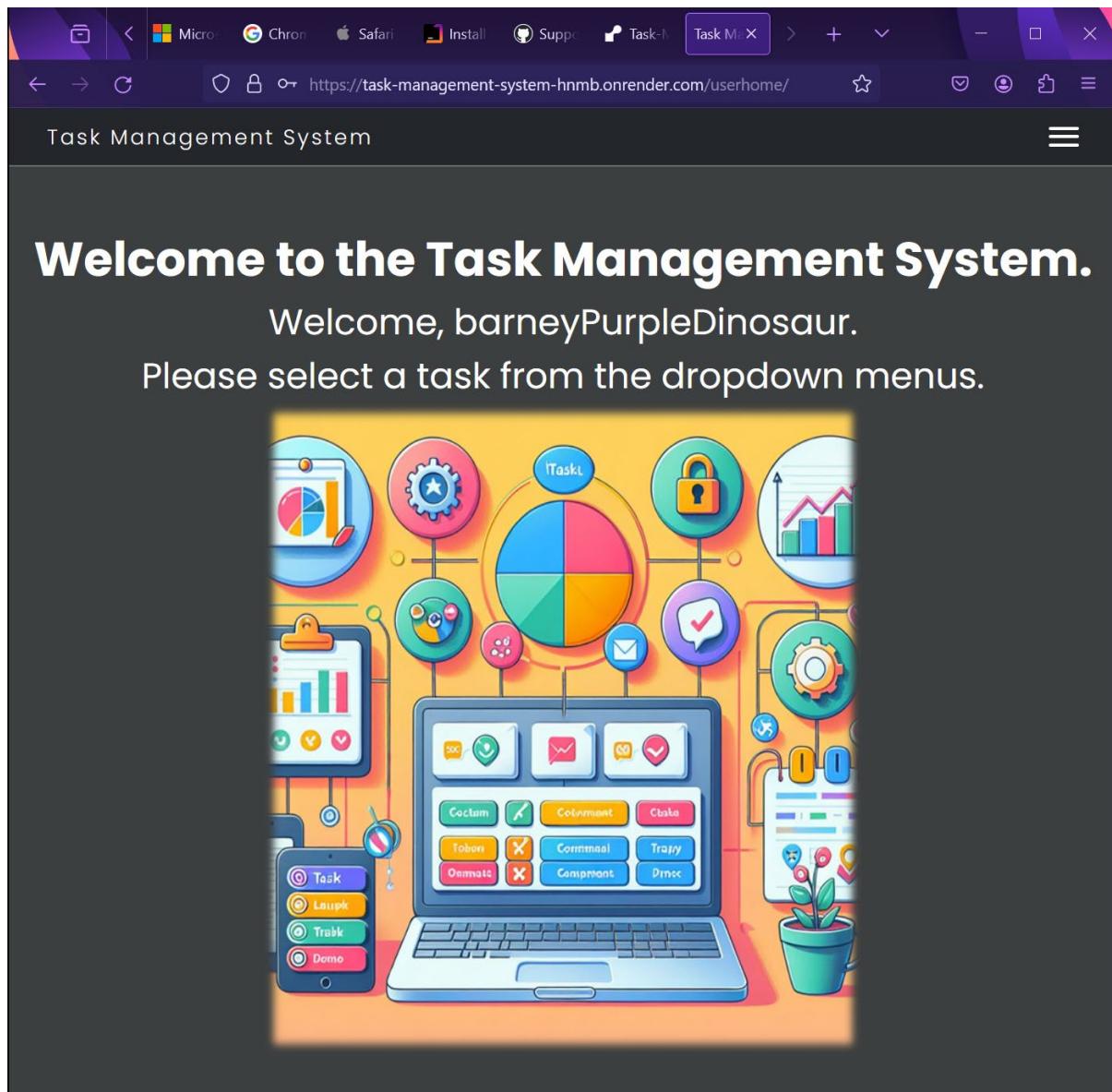


Figure 210 - Task Management System resized on Firefox.

Task Management System	
View Tasks	
TASK ID 1	
TITLE	Test Mobile App
DESCRIPTION	Test the mobile app updated
DUE DATE	2024-02-29
PRIORITY	High
STATUS	Completed
PROJECT ID	1
TASK ID 3	
TITLE	Debug Web App
DESCRIPTION	Test and debug Web Application
DUE DATE	2024-03-01
PRIORITY	High
STATUS	Due
PROJECT ID	2
TASK ID 4	
TITLE	Create C++ Application again
DESCRIPTION	Test the mobile app
PROJECT ID	2

Figure 211 - Task Management System Task table on resized Firefox window.

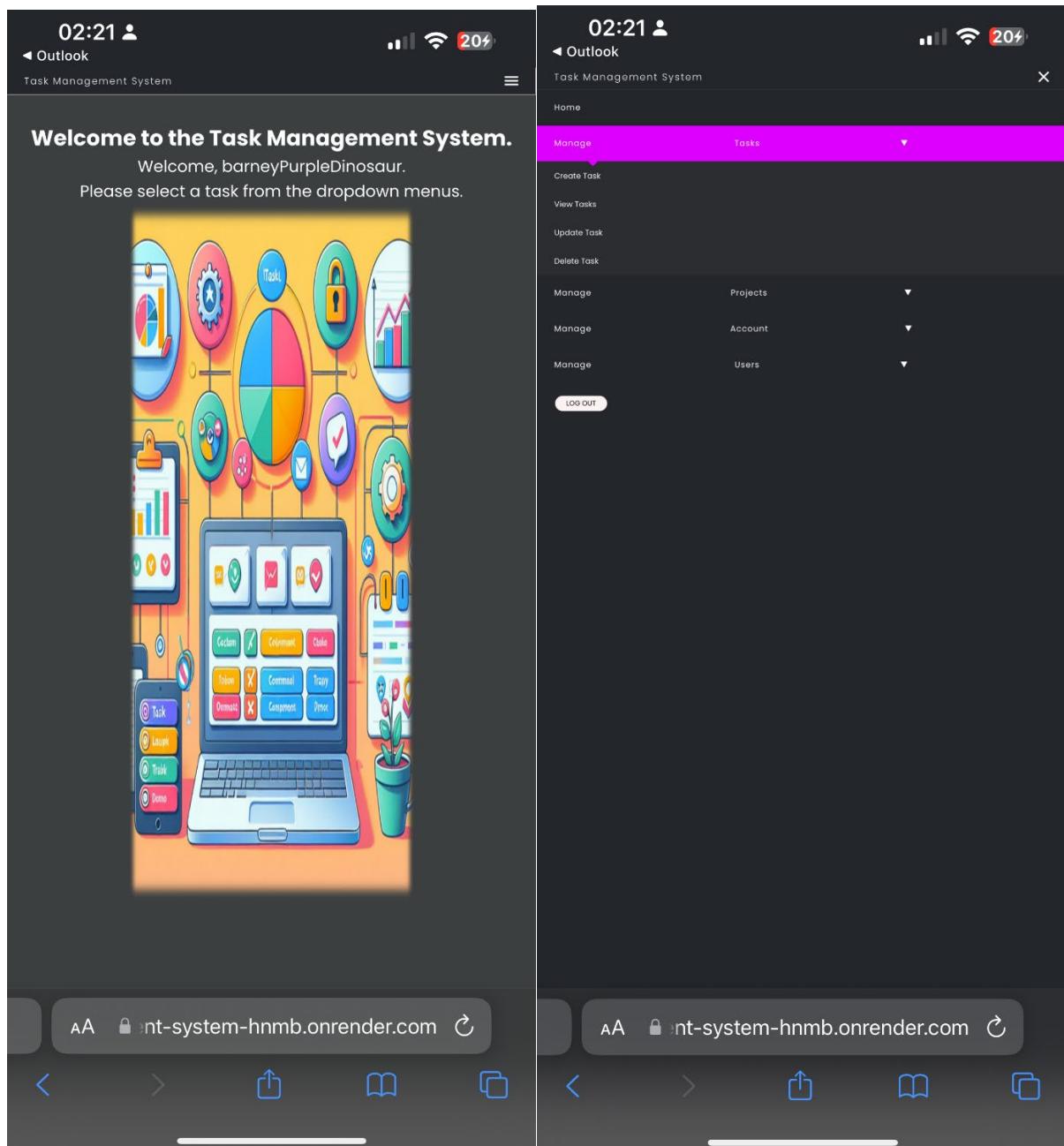


Figure 213 - Expanded Hamburger Menu with navigation (Portrait).

Figure 212 - Task Management System viewed on an iPhone 13 Pro Max (Portrait). Hamburger Menu in the top right corner instead of full navbar.

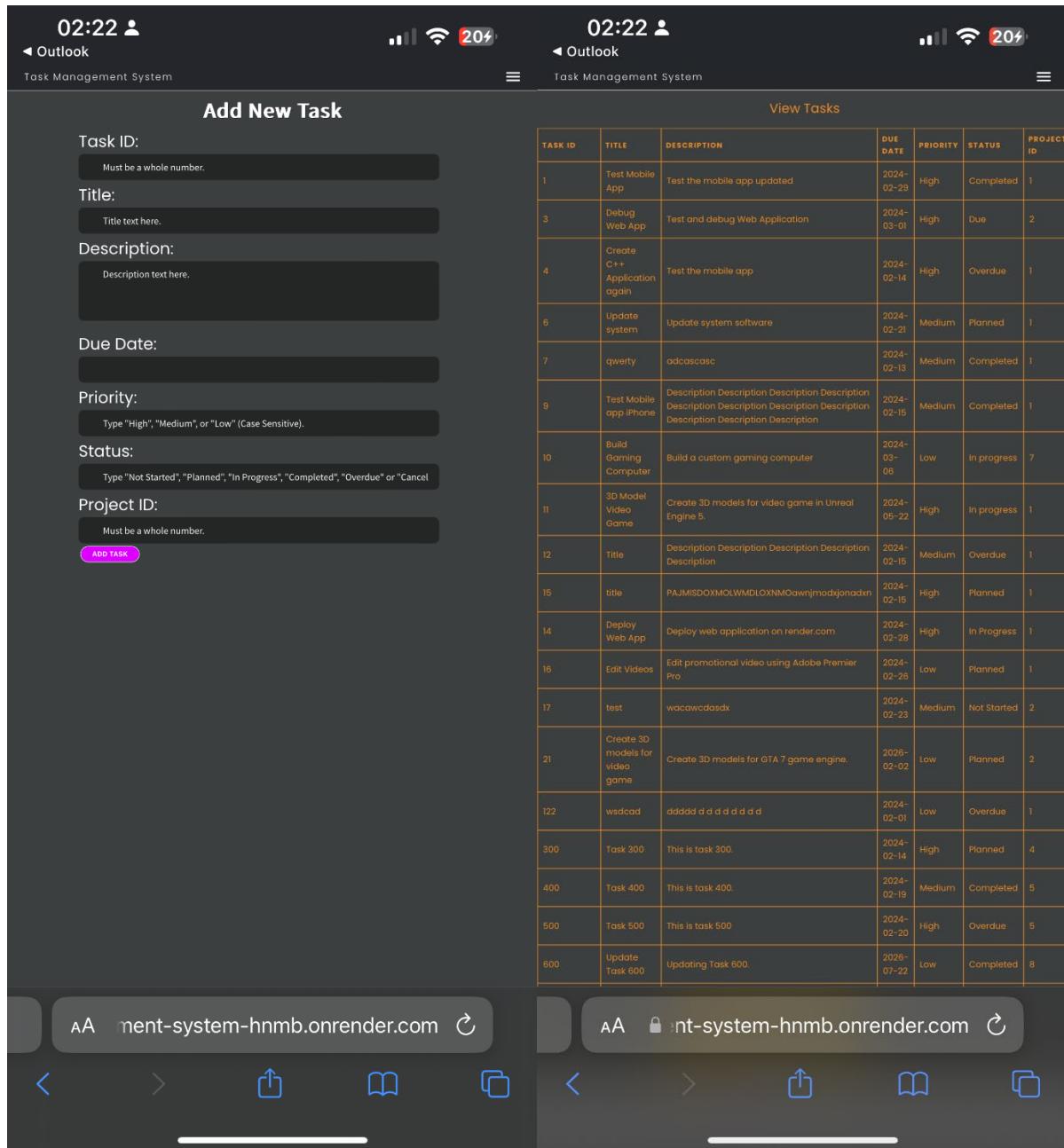


Figure 214 - Form and table viewed on iPhone 13 Pro Max (Portrait).

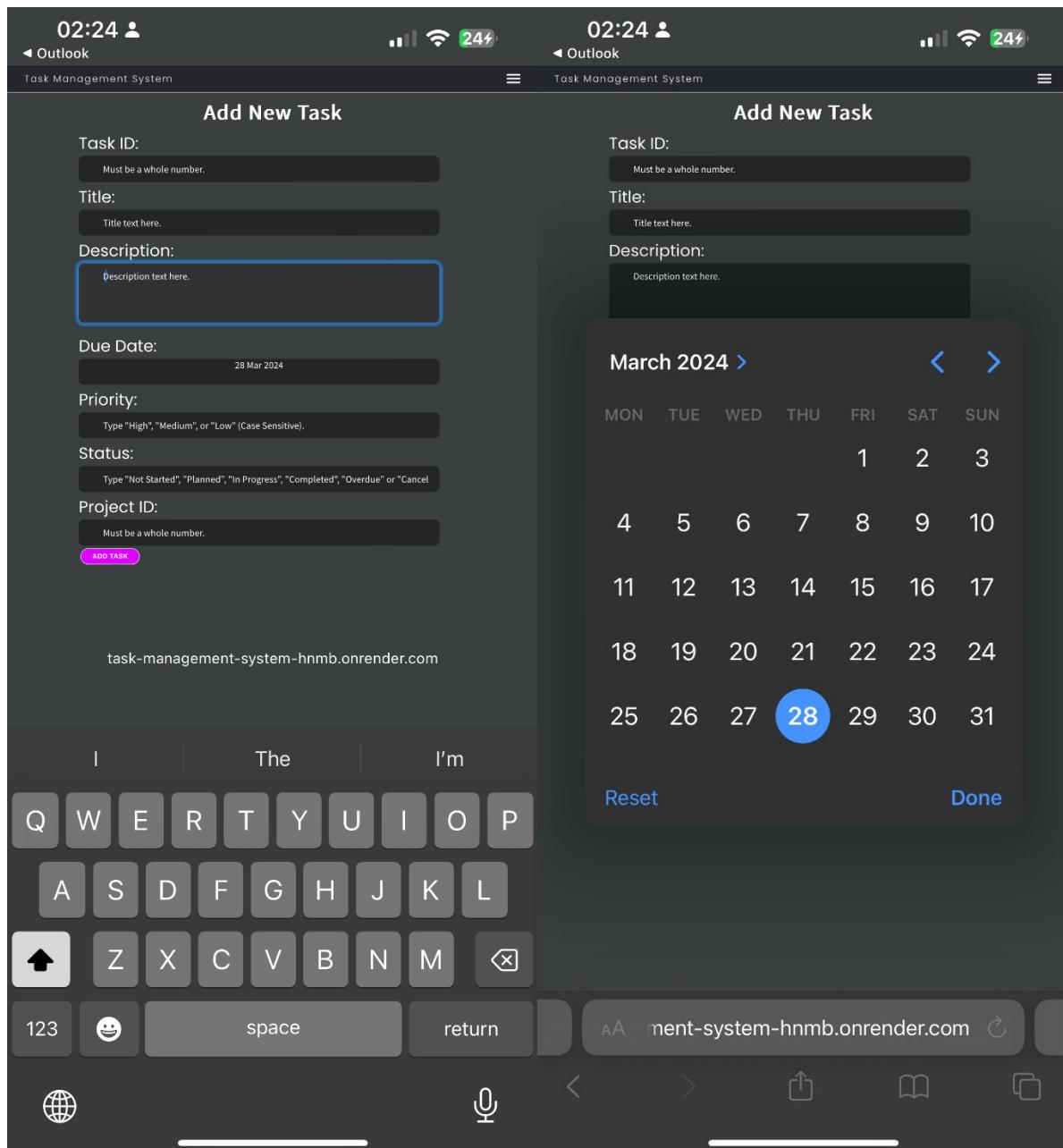


Figure 215 - On-screen keyboard and calendar for Add Task form on iPhone 13 Pro Max (Portrait).



Figure 216 - Task Management System User Home page viewed on iPhone Pro Max (Landscape).

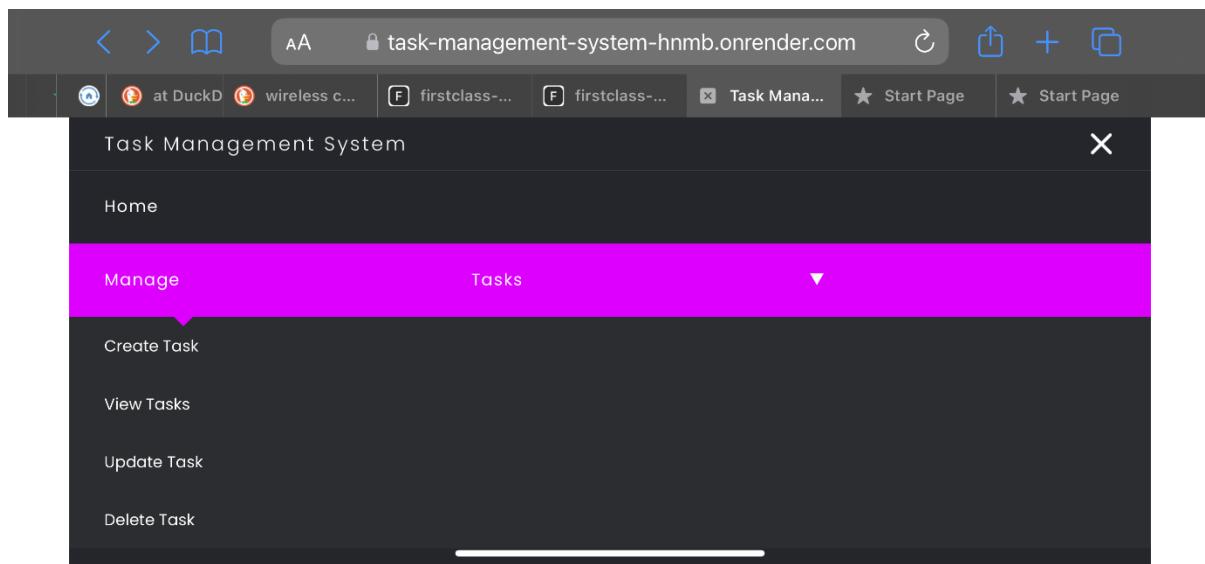


Figure 217 - Expanded Hamburger Menu with navigation (Landscape).

The screenshot shows a mobile web application titled "Task Management System". The main heading is "Add New Task". There are three input fields: "Task ID:" with a note "Must be a whole number.", "Title:" with a note "Title text here.", and "Description:" with a note "Description text here.". The "Task ID:" field has a red border around its input area, indicating an error or validation failure.

Figure 218 - Form viewed on iPhone 13 Pro Max (Landscape).

The screenshot shows the same mobile web application as Figure 218, but with a numeric keyboard displayed over the "Task ID:" input field. The keyboard includes digits from 0 to 9, arithmetic operators (-, /, :, ;, (,), ?, !), punctuation (., , !, ', @, "), and special keys like #+=, space, return, and a microphone icon. The "Task ID:" input field is highlighted with a blue border, and the note "Must be a whole number." is visible above it.

Figure 219 - Form viewed on iPhone 13 Pro Max (Landscape).

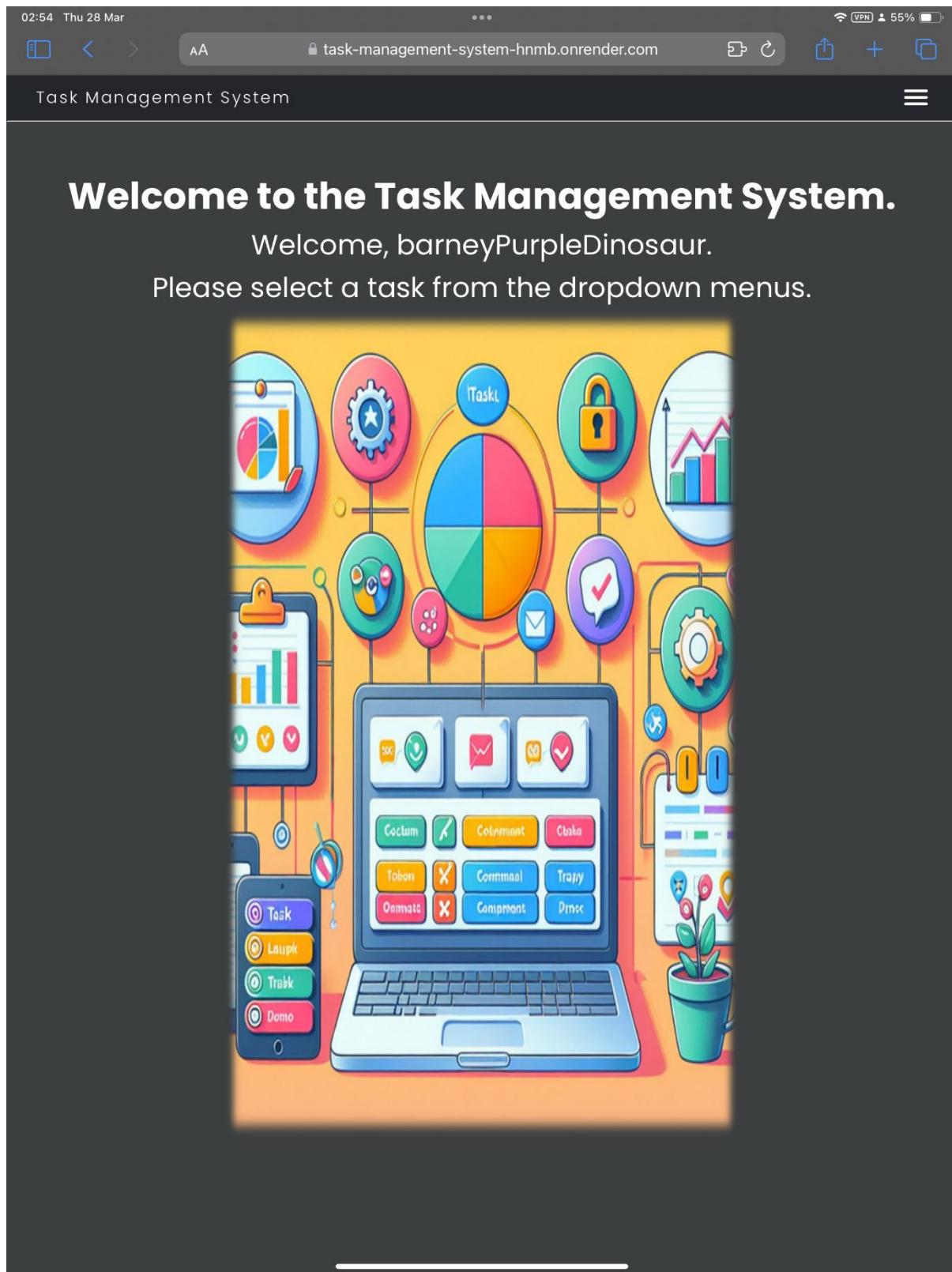


Figure 220 - Task Management System User Home Page viewed on iPad Pro (Portrait).

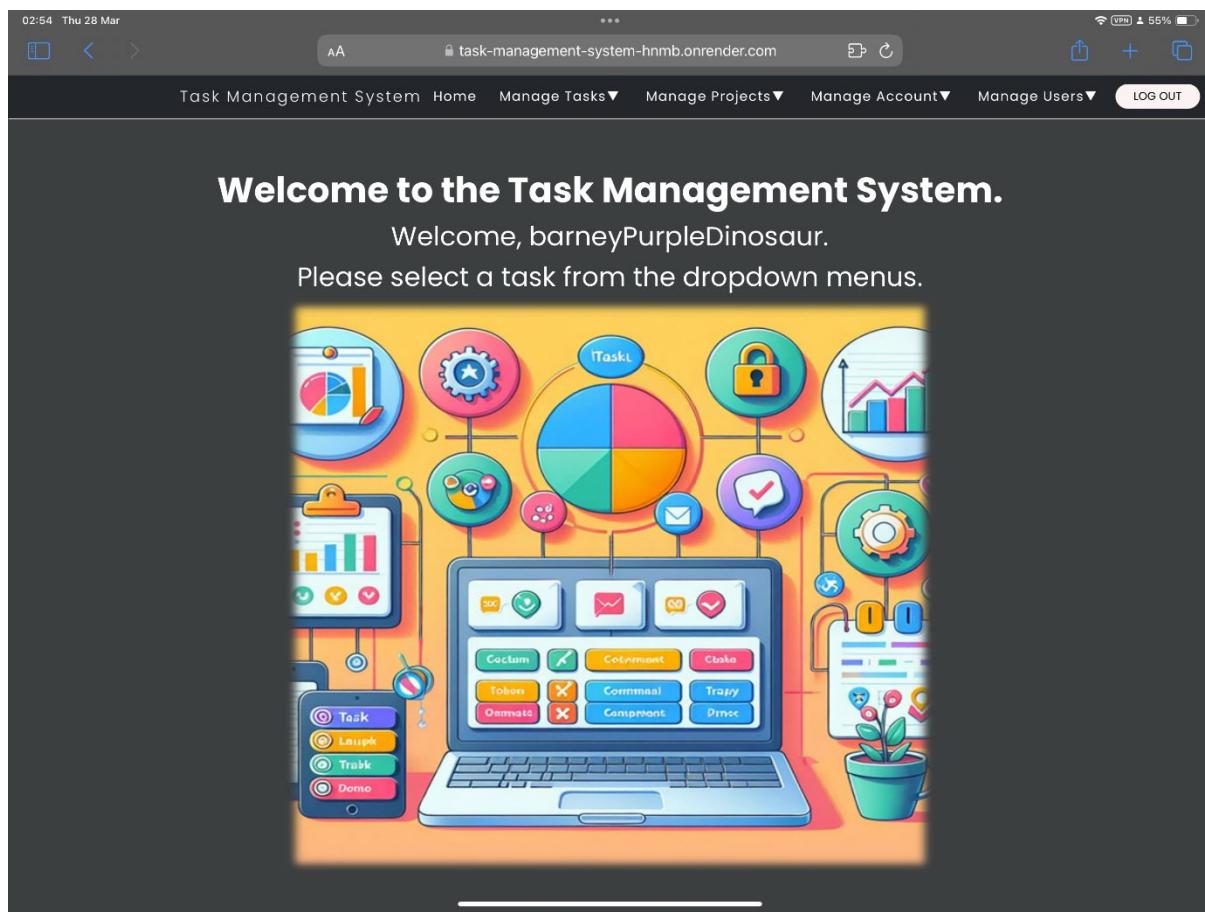


Figure 221 - Task Management System User Home Page viewed on iPad Pro (Landscape).

S275931

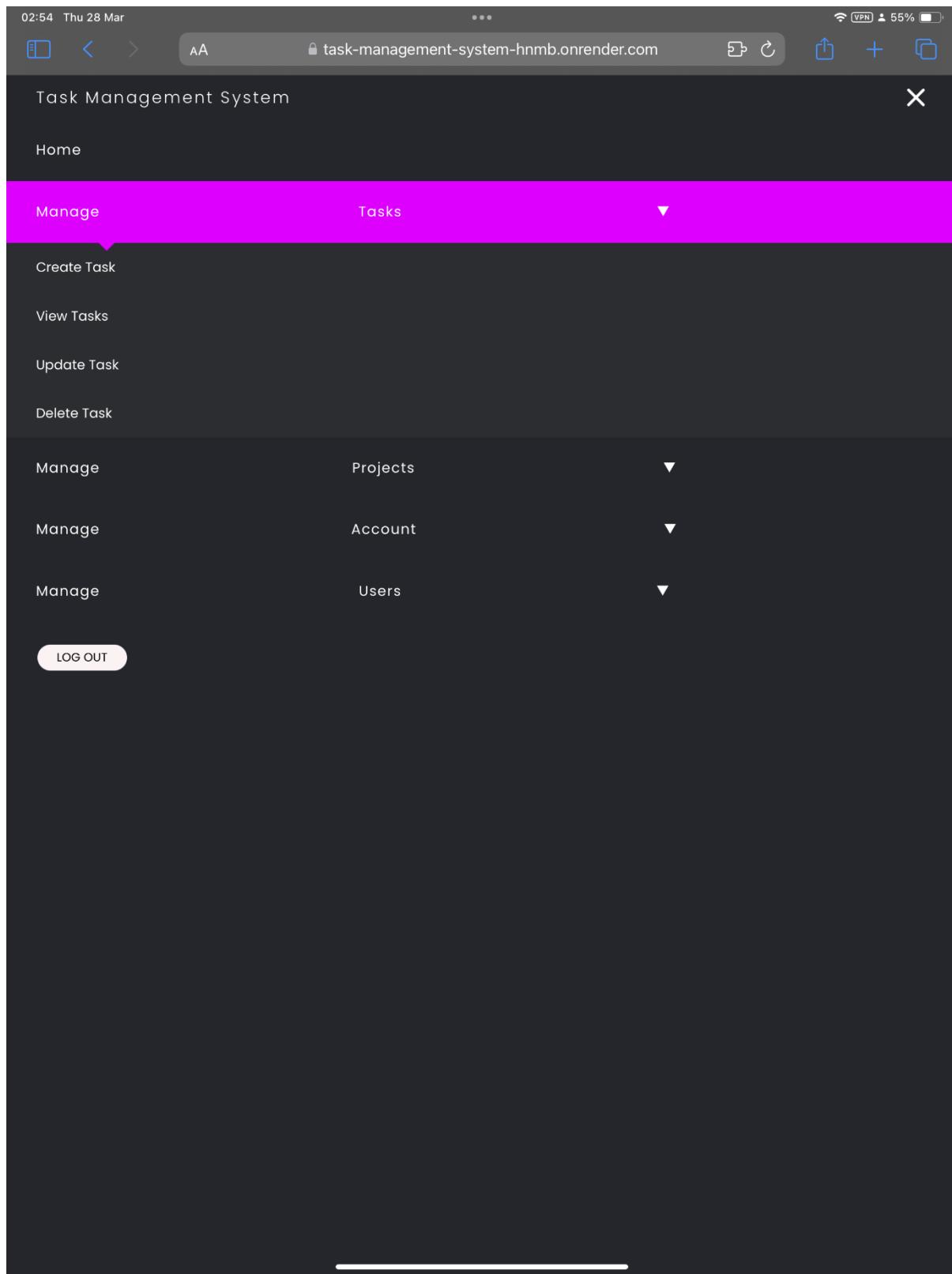


Figure 222 - Expanded Hamburger Menu instead of navbar on iPad Pro (Portrait).

Figure 223 - Table viewed on iPad Pro (Portrait).

Figure 224 - Table viewed on iPad Pro (Landscape).

The screenshot shows a mobile application interface for adding a new project. At the top, the status bar displays the time (02:55), date (Thu 28 Mar), signal strength, battery level (55%), and a VPN icon. Below the status bar is a dark header bar with the URL `task-management-system-hnmb.onrender.com`. The main content area has a dark background and features the following fields:

- Project ID:** A text input field containing the placeholder "Must be a whole number."
- Title:** A text input field containing the placeholder "Title text here."
- Description:** A text input field containing the placeholder "Description text here."
- Assigned Tasks:** A text input field containing the placeholder "List of assigned tasks here."

At the bottom center of the form is a purple button labeled "ADD PROJECT".

Figure 225 - Form viewed on iPad Pro (Portrait).

The screenshot shows a mobile application interface for a 'Task Management System'. At the top, there's a navigation bar with icons for back, forward, search, and refresh, along with the URL 'task-management-system-hnmb.onrender.com'. Below the navigation is a header bar with links for 'Task Management System', 'Home', 'Manage Tasks', 'Manage Projects', 'Manage Account', 'Manage Users', and 'LOG OUT'. The main content area has a dark background and features several input fields:

- Project ID:** A text input field with the placeholder 'Must be a whole number.'
- Title:** A text input field with the placeholder 'Title text here.'
- Description:** A text input field with the placeholder 'Description text here.'
- Assigned Tasks:** A text input field with the placeholder 'List of assigned tasks here.'

At the bottom of the form is a pink button labeled 'ADD PROJECT'.

Figure 226 - Form viewed on iPad Pro (Landscape).

Security Features

The Task Management System has several built-in security features to protect data and prevent unauthorized access, use, and misuse.

1. **User Roles** – There are two main user roles: Standard User and Administrator. The Root user created during setup is an administrator. Administrators have full control of the Task Management System while standard users can only add and view tasks, add and view tasks and manage their own account. Standard users cannot update or delete tasks, update, or delete projects, manage other users, or perform administrative tasks.
2. **Passwords** - All accounts are password protected.
3. **Password Requirements** – Passwords must be a minimum of 10 characters, have at least one capital letter, one lowercase letter, one number, one special character and must not be a common password. This makes it more difficult to guess and crack the passwords.
4. **Password encryption** – All passwords are stored securely on the database as SHA3 512-bit hashes. These hashes are rehashed several times to make it near impossible to decrypt them to obtain the password in the event the database gets hacked.
5. **Maximum 3 failed login attempts** – The user is automatically blocked after three failed login attempts by entering the incorrect password. This also helps prevent automated password cracking, guessing and brute force attacks. The user can only be unblocked by another administrator.
6. **Authentication** – Once the user logs in using the correct username and password (which is checked in the database), a cookie containing the user's username and another cookie

containing the session ID are stored in the user's browser. The session ID is randomly generated every time the user logs into the Task Management System and is also stored in the database. The cookie containing the username must match the username in the database and the cookie containing the session ID must match the session ID in the database for the user to be authenticated and granted access to the Task Management System or else an Access Denied error message will appear instead. When the user logs out, the cookies containing the username and session ID are cleared and the session ID is cleared from the database.

7. Sanitisation – the SQL statements executed in the Python code are sanitised to prevent SQL injection attacks.

References

Apple (2024) *Safari Support*.

Available at: <https://support.apple.com/safari>
(Accessed: 28 March 2024).

Choose an open source license (2007) *GNU Affero General Public License v3.0*.

Available at: <https://choosealicense.com/licenses/agpl-3.0/>
(Accessed: 25 04 2024).

Docker, Inc. (2024) *Docker Docs - Install Docker Desktop on Linux*.

Available at: <https://docs.docker.com/desktop/install/linux-install/>
(Accessed: 28 March 2024).

Docker, Inc. (2024) *Docker Docs - Install Docker Desktop on Mac*.

Available at: <https://docs.docker.com/desktop/install/mac-install/>
(Accessed: 28 March 2024).

Docker, Inc. (2024) *Docker Docs - Install Docker Desktop on Windows*.

Available at: <https://docs.docker.com/desktop/install/windows-install/>
(Accessed: 28 March 2023).

Docker, Inc. (2024) *Docker Hub - Develop faster. Run anywhere..*

Available at: <https://hub.docker.com/>
(Accessed: 28 March 2024).

Gibb, T. (2011) *How-To Geek*.

Available at: <https://www.howtogeek.com/73318/how-to-check-if-your-cpu-supports-second-level-address-translation-slat/>
(Accessed: 28 March 2024).

GitHub (2024) *Let's build from here*.

Available at: <https://github.com/>
(Accessed 28 March 2024).

GitHub (2024) *Supported operating systems for GitHub Desktop*.

Available at: <https://docs.github.com/en/desktop/overview/supported-operating-systems-for-github-desktop>
(Accessed: 28 March 2024).

Google (2024) *Chrome browser system requirements*.

Available at: <https://support.google.com/chrome/a/answer/7100626?hl=en>

(Accessed: 28 March 2024).

Jetbrains (2024) *Install PyCharm*.

Available at: <https://www.jetbrains.com/help/pycharm/installation-guide.html>

(Accessed: 28 March 2024).

Jetbrains (n.d.) *Welcome to JetBrains Account*.

Available at: <https://account.jetbrains.com/login>

(Accessed: 28 March 2024).

Kinda Code (2022) *Docker Desktop system requirements (Windows, macOS)*.

Available at: <https://www.kindacode.com/article/docker-desktop-system-requirements-windows-macos/>

(Accessed: 28 March 2024).

Microsoft (2024) *Microsoft Edge supported Operating Systems*.

Available at: <https://learn.microsoft.com/en-us/deployedge/microsoft-edge-supported-operating-systems>

(Accessed: 28 March 2024).

Mozilla (2024) *Firefox System Requirements*.

Available at: <https://www.mozilla.org/en-US/firefox/124.0.1/system-requirements/>

(Accessed: 28 March 2024).

Render.com (2023) *Render - Your fastest path to production*.

Available at: <https://render.com/>

(Accessed: 28 March 2024).