Maintenance Guide Documentation

Project name: TASMU Robotics
Project team members: Muneera Al-Baker, Anastasia Cheypesh, Ahmad Mohamad

Date: 30/3/2023

1. Introduction

This maintenance guide is designed for the web application. Regular maintenance is essential for ensuring the reliability, security, and performance of the application. This guide provides a comprehensive maintenance schedule and step-by-step instructions for routine tasks, troubleshooting, and repairs.

2. Safety Precautions

Always adhere to the following safety precautions when performing maintenance tasks:

- Keep your development and production environments separate.
- Use version control systems (such as Git) to track changes and maintain a history of revisions.
- Test updates and changes in a staging environment before deploying to production.

3. Maintenance Schedule

Task	Frequency
Backup Data	Weekly
Update Dependencies	Monthly
Monitor Logs	Daily
Performance Optimization	Quarterly

4. Routine Maintenance Procedures

4.1. Backup Data

- Schedule automated backups of your database and application data.
- Store backups in a secure, offsite location.
- Periodically verify the integrity of the backups.

4.2. Update Dependencies

- Check for updates to Flask and other dependencies using package managers like pip or conda.
- Review release notes to understand the impact of updates on your application.
- Test updates in a staging environment before deploying to production.

4.3. Monitor Logs

- Regularly review log files for errors, warnings, or unusual activity.
- Set up log monitoring tools to send alerts for critical events.
- Address issues as they arise to maintain the stability of the application.

4.4. Performance Optimization

- Monitor application performance metrics, such as response times and resource usage.
- Identify and address performance bottlenecks by optimizing database queries, caching, or load balancing.
- Regularly review and update your application's architecture to meet growing demands.

5. Troubleshooting and Repairs

5.1. Server Issues

Verify that your server is running the latest version of the operating system and all required software packages.
Monitor server resources, such as CPU, memory, and disk usage.
Address server issues by scaling resources or migrating to a more powerful server, if necessary.

5.2. Application Errors

Review error logs to identify the root cause of issues.
Debug and fix application code, following best practices for web development.
Test and deploy fixes to the staging environment, then to production.

5.3. Security Vulnerabilities

Keep your application's dependencies up to date to minimize security risks.
Regularly review your application code and configuration for security vulnerabilities.
Implement security best practices

6. New Features

Plan and design the new feature:

- Define the purpose and desired functionality of the new feature
- Determine how it will integrate with the existing application

Set up a development environment:

- Create a new branch in your version control system (e.g., Git) to work on the new feature without affecting the main application
- Ensure your development environment is isolated from the production environment, using tools like virtual environments or Docker containers

Implement the new feature:

- Write the code for the new feature, following best practices for organization, readability, and documentation
- Test the new feature thoroughly, both manually and using automated testing tools, to ensure it works as intended and doesn't introduce new issues

Integrate the new feature:

- Merge the new feature branch into the main branch, resolving any conflicts that arise
- Update the application's documentation, including the README file, to reflect the new feature's functionality and usage
- Deploy the updated application to the production environment, following your established deployment procedures

Monitor and maintain the new feature:

- Monitor the application logs and user feedback to identify any issues or performance bottlenecks related to the new feature
- Make necessary updates or optimizations to ensure the new feature continues to function well within the application.