# Maintenance Guide – divplusplus

1. System Requirements

* **Operating System:** The server is hosted on Windows
* **Software Requirements:**
  + Backend: Node.js
  + Frontend: React
  + Database: MongoDB
  + Testing: Jest
* **Hosting Requirements:** The website is hosted on Microsoft Azure.

2. Codebase Management

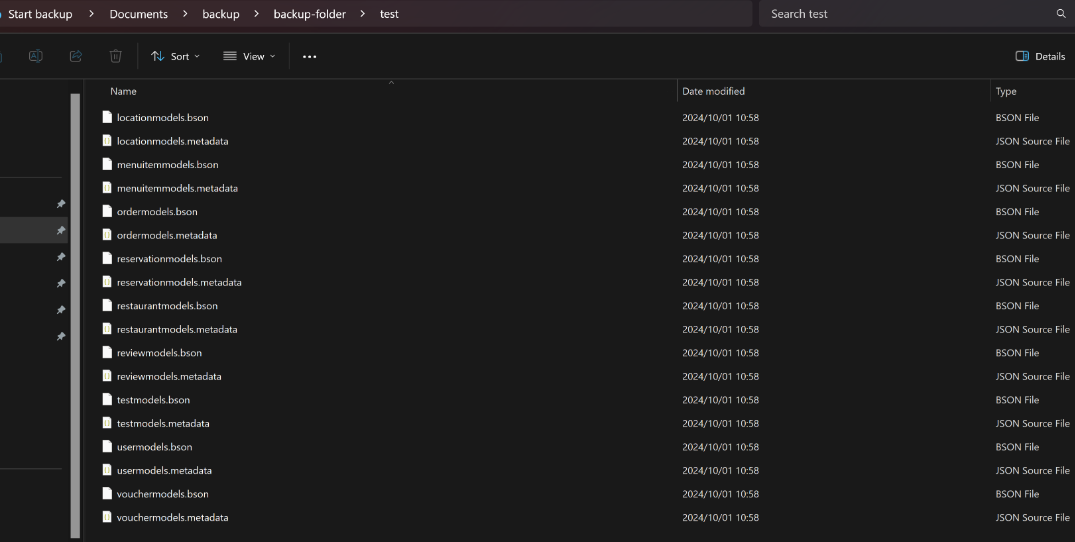
* **Version Control:** Make sure a GitHub repository is used to log any code modifications, using CICD and we created a separate GitHub Repository for Frontend and Backend.
* **Branching Strategy:** We had a main and testing branch on both repositories, where testing was done and pushed to the testing branch before pushed to the main branch.
* **Deployment Process:**

**Development Workflow:**

* Developers work on new features or bug fixes in separate branches so that the workflow is not interrupted.
* Once the changes are finalized and tested locally, developers create a pull request to merge the feature branch into the main or production branch.
* Developers push the updates to the GitHub repository once the code has been merged into the main branch.
* Azure is configured to automatically detect changes in the repository.
* Azure pulls the updated code from GitHub and redeploys the application, so that the live website is updated.
* When Azure detects a push to the main branch, it starts the build process.
* During the build, the following occurs:
  + Application dependencies are installed.
  + The frontend (React) and backend (Node.js) are compiled or bundled.
  + Static assets (HTML, CSS and JavaScript) are optimized for production.
  + **Environment management**, we had a development environment where we built and tested new features then we moved to a staging environment where we ran the application locally to test it before it is deployed and then the production environment where this is the live environment where the public has access to it.

3. Backup Procedures

* **Database Backup:**
  + Frequency: Our database is manually backed up to google drive due to us needing to pay fees to automatically back it up on MongoDB.
  + Tools: Google Drive.
  + Code - mongodump --uri "mongodb+srv://2460755:qg2XJWBsvqW5yfJt@sdpcluster.vux5q.mongodb.net/" --out documents/backup/backup-folder



* **Codebase Backup:**
  + We make sure that the codebase is backed up using the GitHub repository.

4. Security and Patching

* **Security Patches:**
  + We have made sure that we monitor the platform libraries and frameworks to make sure they are always up to date to minimize vulnerabilities.
* **User Authentication:**
  + We regularly review the authentication mechanism namely auth0 to ensure access control.

5. Database Management

* **Data Integrity Checks:**
  + Periodically check for data integrity issues.
  + Perform regular audits of the meal credits, reservations, and feedback data.

6. Regular Maintenance Schedule

* **Daily:**
  + Monitor server and database logs for unusual activity.
* **Monthly:**
  + Perform database optimization tasks.
  + Review code and dependencies for updates or refactoring needs.
* **Quarterly:**
  + Perform a security audit, including reviewing user access and vulnerability testing.
  + Review feedback from users to improve the user experience.

7. User Manual

* + We have created a user manual to explain how to use our application.