# Peregrine Service Plan of Attack

Our goal is to create a WCF service for message logging that can be deployed on a Microsoft server running IIS. The service needs to be able to create and manage the logging database as well as provide SOAP methods for accessing that database. It is essential that our service has no downtime and is only performing a task when a request is made to it or it is performing database maintenance. It is also very important that the service can handle multiple connections to it.

Each log message should be associated with a “Process” that it came from. Messages need to contain useful information such as a timestamp and a priority. The service is responsible for inserting, retrieving and deleting these messages.

“Processes” can also contain some number of “Jobs”. The service needs to track these optional jobs in order to provide update information about the same job at a later time.

We will use Microsoft Visual Studio 2010 to develop the service in C#. We will also use review board and git to help up us with our version control. To test deployment, we will use Microsoft Server 2008 and have it run IIS. For automated testing, we will use Hudson. Automated tests might include sending in a collection of requests to perform actions on the database or performing a timed request to make sure the service isn’t hanging too long somewhere.

See the database plan of attack for more detailed information about the “Process”, “Message” and “Job” objects.