

## v1.0.0 Design

### backup

- **backup** in directory 'script' will recursively zip its working directory and copy it to a passed remote user, host, and directory (1).

### archive

- **archive** will be a tested Go package that has a single function **List** that accepts a directory and returns all its zip archives with file-sizes and upload-times (2).

### Server

- The server will be able to be configured with a port and an archive directory (2).
- A handler at path '/script' will read 'script/backup' and replace the remote user, host, and directory with the server's configured values before returning the script (2).
- A static file handler will serve static files from the directory 'static' (2).
- A static file handler will serve archives from the directory 'archive' (2).
- A home page handler will inject all archives in the archive directory into a template that displays each archive with its download link, file-size, and upload-time in a table ordered by the most recent upload time (2).
- The home page will also display a link to download the backup-script (2).
- Templates will be stored in a directory called 'tmpl' (2).
- The template, archive directory, and script will be cached (2).

### Utilities

- **deploy** will deploy the server to a passed remote user and host (3).
- **copy\_assets** will copy 'script', 'static', and 'tmpl' to a passed remote user, host, and directory (3).