**Creating Alpha-0 release**



1. Create backup on tm3ror development directory before striping directory for alpha test release [DONE]
2. Strip local directory to reduce size of installation image on alpha test server

* remove .bu and test html.erb files in views subdirectory [DONE]
* remove no longer used javascript files from public/javascripts [DONE]
* remove no longer used css files from public/stylesheets [DONE]
* pull out all .png files from public/images/categories into separate directory outside of tm3ror [DONE]
* pull out all directories of .png files from public/images/products into a separate directory outside of tm3ror [DONE]
* [ remove currently unused public/images/medium directory ]

1. Create git commit of stripped directory in master branch of git [DONE]
2. Push to github [DONE]
3. Set up trumedical subdirectory within amazon/heroku account
4. Reset remote repository declaration in tm3ror to point to that subdirectory
5. Check GitIgnore to make sure don't attempt to move either git history, or sqlite3 database
6. Attempt to move the rest of the directory ( including home, category, and product images, as well as excel files for building the database tables )
7. If compiled slug is within size limits, then migrate Sqlite database from local server to a new Postgres database on the alpha test server, and test that application comes up on alpha test server.

* run scripts to correct format of dates in timestamp columns of tables in database
* fix format of any other columns that cannot be migrated to Postgres
* attempt to invoke the application
* examine logs to determine what is preventing it from coming up
* fix these issues

1. Alternatively, if compiled slug exceeds 100MB maximum, then
2. move all images ( home, category, product, etc.) to an Amazon S3 bucket, setting permissions and expires headers
3. rebuild page templates to reference images from Amazon S3 servers — creating a new development version of the site with its own git repo history
4. test locally from development directory
5. strip images out of /public/images/... subdirectories
6. git commit smaller repository
7. push to heroku and build runtime image
8. migrate the database from local server to alpha test server ( step #9 above)
9. test that site comes up on heroku/amazon, pulling images from amazon S3.
10. Rebuild XML site map, install on both development and alph test versions of the site and request reindex of the alpha version of the site from Google