3 Rules

The 3 rules

Used by the project admins in the "Recruitment Ceremony". A one line set of maxims on how members of the Tiki Community collaborate.

I wish to follow the 3 rules, and also to convince anyone in Tiki community to respect those simple rules:

In brief

1. **Respect the Environment**
   Tiki is both a software project and a community. The mix of both, contextualized on the Internet and in real life, is called *The Environment*. Any change in Tiki should take in account its effect on the entire Tiki community and should allow for a balanced evolution with respect for the people and organizations that use it. Please make sure any code you commit respects the LGPL license and that you allowed to share it. Do not duplicate code or features, as this just creates more work for everyone to maintain. Reuse and extend code instead, and ask for help if you are unsure how, or unsure if there is similar code lying around.
2. **Commit early, Commit often**
   SVN is the central point in Tiki collaborative development. Commits should be frequent, even in the early stages, to offer an opportunity for interaction between contributors. Early commits provide more chances for feedback and for the community to start to assimilate the ideas you propose. "Upstream first" is is the cheapest, most sustainable way to innovate on an open source platform.
3. **Make it Optional**
   Tiki is used in many smaller contexts and its modularity is key to adoption in those areas. Help preserve this flexibility by making your changes optional whenever possible, accessible for tuning to the admin at least, via admin panels. This is also good for security as if a security vulnerability is discovered, feature can be deactivated.

Verbose mode

1. **Think about other users**
   Tiki is both a piece of software and a community of people. This combination means that we invite you, as a member of the community, to think not only about the code but also the wide variety of people who use Tiki every day. Consider your proposed changes in this context. We believe that a careful, thoughtful, and highly collaborative approach is a way to maintain respect for both the code and the people who depend on it. Rather than seeing Tiki as a game, we invite you to see Tiki as a manner of producing change. Recognize that your code could affect the lives of people.
2. **Share Early, Share Often**
   If you have an idea for an improvement, new feature, performance enhancement, or anything else of that nature, be quick to share it. Be proud of your idea and get it out there on the e-mail list or IRC channel. Be open to questions and ideas that others may have. As you work out your ideas and implementations, share your progress and approach often. Ask for advice and feedback. There are many smart and dedicated people in the Tiki Community who love to help. Documenting what you are doing on Tiki.org keeps others up to date with changes. You are encouraged to create a wiki page for your idea so others can comment. Create a showcase site to show off your work in progress. Once it basically works and the community likes it, commit your code to SVN. Yes, it may be imperfect, nevertheless by following the maxim of Release Early, Release Often others are more able to help with development and debugging.
   One BIG caution: Don't commit sweeping or wide-reaching changes to SVN until there is community consensus or at least approval from one or more of the project administrators. They are those who have that designation in the list of developers. Checking with others is the right way to develop code

and helps us to avoid really screwing up other people's lives and projects. When in doubt, communicate! This could be on IRC, by email, or some other agreed upon method. For sweeping changes, it's encouraged to use an Experimental Branch, which can be merged to trunk later.

A caution about the BIG caution *: I feel it is impossible to reach consensus without effective code. Asking before is a matter of gathering information, not getting prior acceptance of something. Good decisions can provide bad implementations and in such cases, there is a difficulty in correcting what was mutually agreed beforehand (without really knowing). Of course, that only applies to experienced coders **who are supposed to know what they are doing**. People who learn to code need to be particularly cautious. You decide whether this is for authoritative reasons or as a means of obtaining wisdom (if available). That's my 2 cents. — mose*

3. **Make It Optional**
   Tiki is used in the real world by *many* people for *many* different uses. Try to avoid forcing new features on everyone. Allow new features to be tuned and configured by the site admin, and, if at all possible, allow it to be turned off. At the very least, make sure that the default config doesn't change Tiki's behavior.

More good things to do

- Register to the SVN & Developers mailing list. We need to have a way to join you if you have some comments to do on your commits.
- Hang out on IRC or XMPP.
- Read Hello World.
- Follow DevTips.

Alias

3Rules | Ceremony

Things to discuss

- "default config doesn't change Tiki's behavior." -> "default config is the most common/expected" (This is subjective of course, which is why we have http://profiles.tiki.org).
- Clarify expectation about backward compatibility (How and when to break things)