Tool Co-op

Project Overview

This project aims to build a Tool Co-op website for a tool sharing service of a company.

The website will be a platform for the employees of the company to manage the lending and returning of tools. The goal of the website is to automate and eliminate error while lending and returning tools. The website will have a database for all tools in operation and will clearly label tools that are checked in and checked out.

Team Organization

Our team is a self-organized team. Team members will communicate as a team whenever possible to ensure that the group is working toward its goals. Members are encouraged to take on tasks without waiting for them to be assigned. Frequent review of the highest priority tasks will happen often to reevaluate current goals and progress towards them. Team members will take responsibility for their work, and track and report on their own progress. Team members will be vocal if problems arise to avoid delay on task completion and to maintain team synergy. Members will adopt the principles of 'egoless programming' and will be receptive and communicative towards feedback from other team members. Team members should always voice their criticisms constructively and maturely.

Software Development Process

The development will be broken up into five phases. Each phase will be a little like a Sprint in an Agile method and a little like an iteration in a Spiral process. Specifically, each phase will be like a Sprint, in that work to be done will be organized into small tasks, placed into a "backlog", and prioritized. Then, using on time-box scheduling, the team will decide which tasks the phase (Sprint) will address. The team will use a Scrum Board to keep track of tasks in the backlog, those that will be part of the current Sprint, those in progress, and those that are done.

Each phase will also be a little like an iteration in a Spiral process, in that each phase will include some risk analysis and that any development activity (requirements capture, analysis, design, implementation, etc.) can be done during any phase. Early phases will focus on understanding (requirements capture and analysis) and subsequent phases will focus on design and implementation. Each phase will include a retrospective.

Phase	Iteration
1.	Phase 1 - Requirements Capture
2.	Phase 2 - Analysis, Architectural, UI, and DB Design
3	Phase 3 - Implementation, and Unit Testing
4	Phase 4 - More Implementation and Testing

We will use Unified Modeling Language (UML) to document user goals, structural concepts, component interactions, and behaviors.

Communication policies, procedures, and tools

1. Team members must work on highest priority tasks.
2. Assign a team member to Pull Request for Code Review.
3. Code Reviewers must read code and test code before approval.
4. Approval of code changes is required before merging into master.
5. Primary form of communication will be in class and in designated group meetings. All out-of-class communication will be managed through Slack.
6. All code will be located in a Github Repository: https://github.com/connordibble/Tool-Co-op.git
7. Code will be written using the Django framework accompanied by Python.
8. Tasks will be selected from a priority list found in the Requirements Gathering Document.
9. Slack channel is used to know what tasks are currently being worked on.
10. Unit tests must be written for every part of code that is not UI

Configuration Management

See the README.md in the Git repository.