

# Практика 1: Проектирование системы контроля версий

Юрий Литвинов

y.litvinov@spbu.ru

В командах по 2-3 человека требуется спроектировать систему контроля версий, представляющую из себя консольное приложение и умеющую:

- commit с commit message, датой коммита и автором;
- работу с ветками: создание и удаление;
- checkout по имени ревизии или ветки;
- log — список ревизий вместе с commit message в текущей ветке;
- merge — сливает указанную ветку с текущей.

При этом код системы должен позволять себя использовать как библиотеку, но предполагается также наличие консольного интерфейса.

Что надо сделать:

- диаграмму компонентов с классами, пока как умеете — про формальные аспекты рисования диаграмм начнём говорить на следующей неделе;
- текстовое описание не требуется, поясняйте непонятные моменты в комментариях на диаграмме;
- нельзя подсматривать в Git Book и другую архитектурную документацию систем контроля версий (там всё написано, хотя конкретно git нельзя назвать примером хорошей архитектуры, и у нас про это ещё отдельная пара будет).

Решения в духе «куча классов, хаотично соединённых стрелками» не пойдут, подумайте над некой высокоуровневой структурой.

Обратите внимание на следующие вещи.

- Как представляются файлы, коммиты, ветки, репозиторий?
- Как выполняется компрессия и выполняется ли вообще? Насколько просто получить текущую, предыдущую, произвольную версии?
- Каков жизненный цикл файла?
- Как выполняется работа с файловой системой?
- Как выполняется работа с пользователем? Как представляются команды?