

# VERSION CONTROL

## WITH GIT AND SOURCEKETTLE

Tom Blount

Sildes available at: <http://tomblount.co.uk/version-control>



# INTRODUCTION

- What is version control?
- What is git?
- What is SourceKettle?

# VERSION CONTROL

- Also called source control, revision control, source management...
- Boils down to: keeping your work safe
- Can revert to previous versions
- Prevents access conflicts

# VERSION CONTROL

- Bad examples:
  - Memory sticks
  - "Biscuit tin"
- Better examples:
  - Cloud Services (Dropbox)
- Good examples:
  - SVN
  - git

# VERSION CONTROL

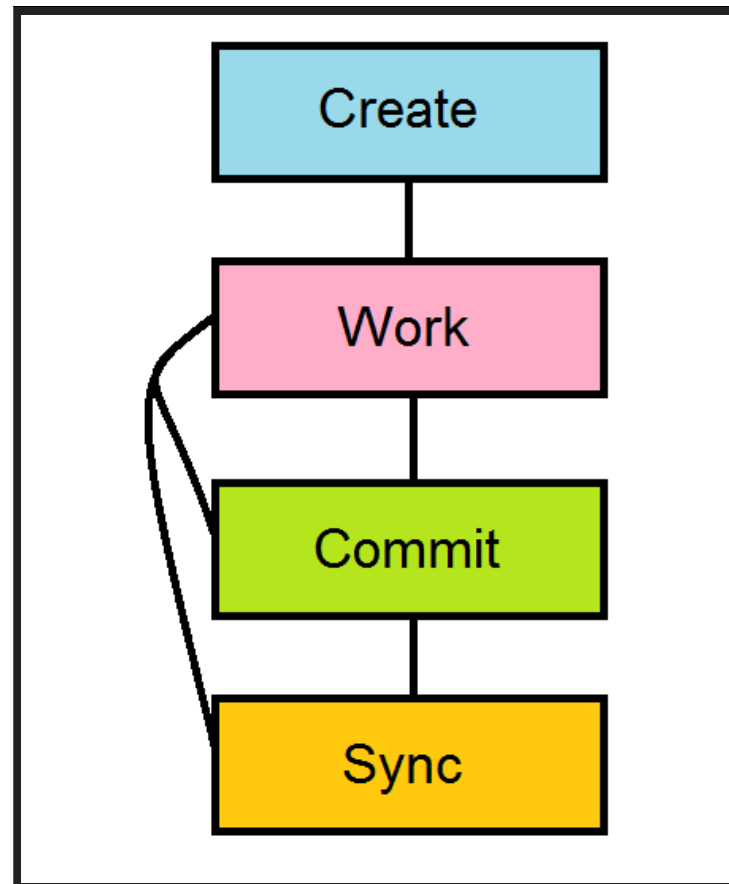
- *Commit* your work to a *repository*
- Repositories maintain a history of work on a project
- Repositories can be *local* or *remote* (or both)
- *Distributed* repositories are shared across multiple machines

# GIT

- Command-line tools:
  - Windows: <http://msysgit.github.io/>
  - Linux: `sudo apt-get install git`
- GUIs:
  - [TortoiseGit](#) (Windows)
  - [GitX](#) (Mac)
  - Also plugins for IDEs such as Eclipse

# GIT

Overview:



# GIT

## Setting up git:

```
Tom@ECS ~/Documents/SEG_Project  
$ git config --global user.name "Tom"  
$ git config --global user.email tb12g09@ecs.soton.ac.uk
```



# GIT

Initialising a (local) repository:

```
Tom@ECS ~/Documents/SEG_Project  
$ git init  
Initialized empty Git repository in C:/Users/Tom/Documents/SEG_Projec
```

# GIT

## Accessing an existing project:

```
Tom@ECS ~/Documents
$ git clone git@sourcekettle.ecs.soton.ac.uk:projects/SEG_Project.git

Cloning into 'SEG_Project'...
remote: Counting objects: 4051, done.
remote: Compressing objects: 100% (2824/2824), done.
remote: Total 4051 (delta 1170), reused 4051 (delta 1170)
Receiving objects: 100% (4051/4051), 2.10 MiB | 314.00 KiB/s, done.
Resolving deltas: 100% (1170/1170), done.
Checking connectivity... done.
Checking out files: 100% (1384/1384), done.
```

You'll need to set up a **Public Key** to be able to do this!

# GIT

## Checking the status:

```
Tom@ECS ~/Documents/SEG_Project (master)
$ git status
# On branch master
#
# Initial commit
#
nothing to commit (create/copy files and use "git add" to track)
```

# GIT

## Creating and adding a file:

```
Tom@ECS ~/Documents/SEG_Project (master)
$ touch README
$ git add README
$ git status

# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   README
#
```

# GIT

## Committing a file:

```
Tom@ECS ~/Documents/SEG_Project (master)
$ git commit -am "Added README file"

[master (root-commit) d616bbe] Added a README
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README
```

# GIT

Connecting to a remote repository:

```
Tom@ECS ~/Documents/SEG_Project (master)  
$ git remote add origin git@sourcekettle.ecs.soton.ac.uk:projects/SEG_Project
```

You'll need to set up a **Public Key** to be able to do this!

# GIT

Pushing/pulling changes to/from the remote server:  
(You need to be on the VPN to use this)

```
Tom@ECS ~/Documents/SEG_Project (master)
$ git push -u origin master

Counting objects: 3, done.
Writing objects: 100% (3/3), 215 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@sourcekettle.ecs.soton.ac.uk:projects/SEG_Project.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin

$ git pull
Already up-to-date
```

**Hint:** always do a *pull* before you *push*!

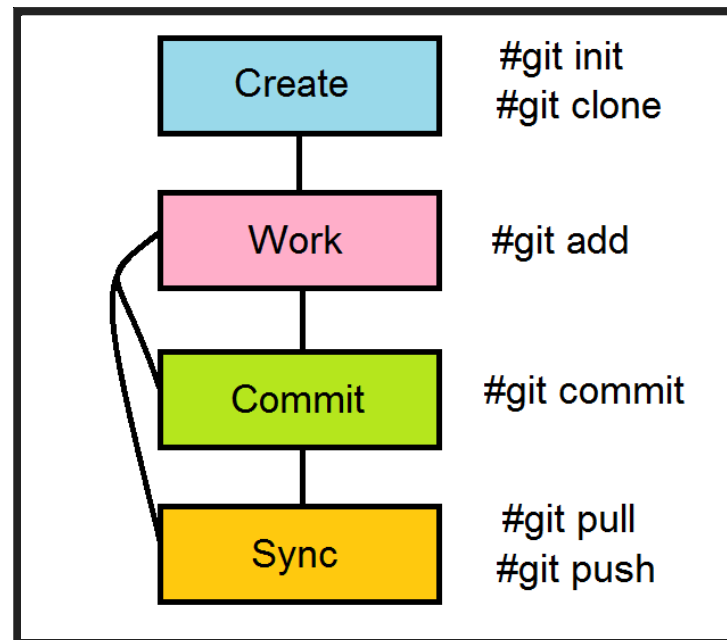
# GIT: CONFLICTS

- Git will try to handle people modifying the same file
- Git will show a conflict if people modify the same line of the same file
- To fix: edit the file to the "correct" version and recommit



# GIT

## Overview (again):

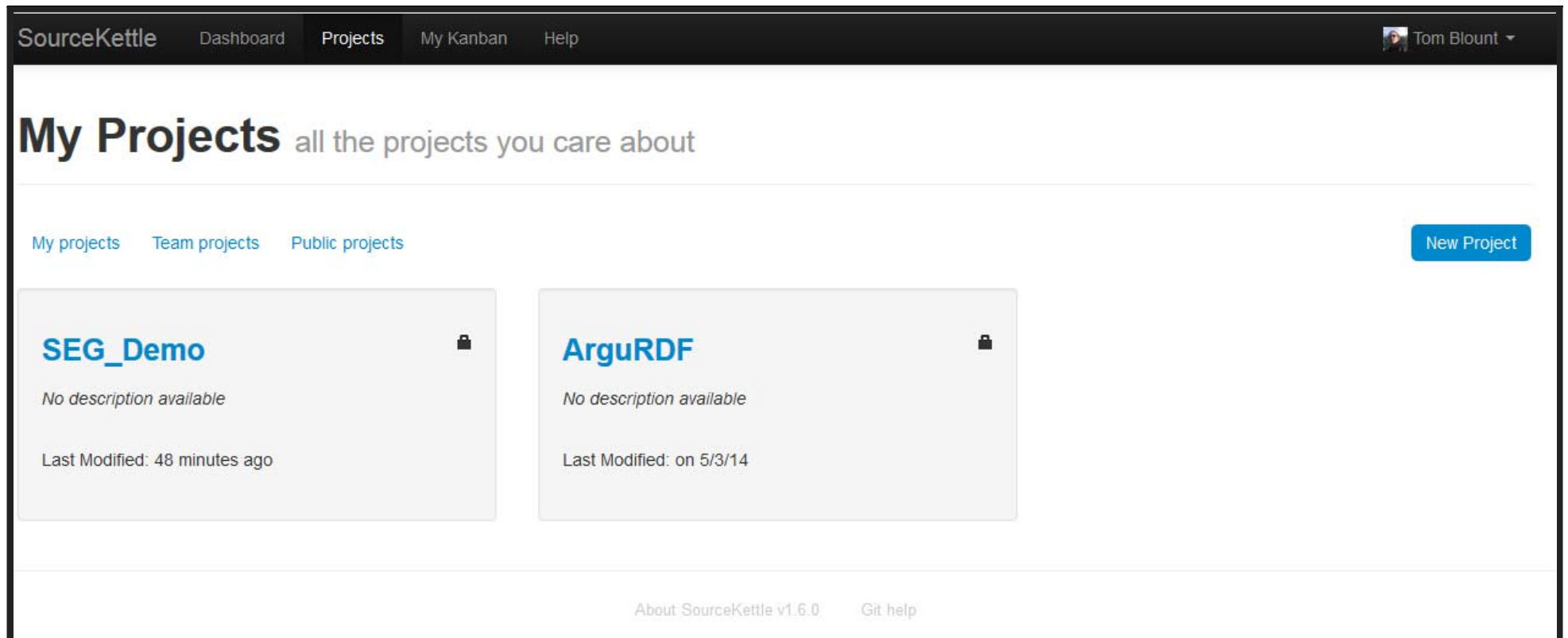


# SOURCEKETTLE

- <https://sourcekettle.ecs.soton.ac.uk/>
- Stores your code repositories on a server in ECS
- Dynamic Task Tracking
- Time Tracking
- Kanban/Timesheets/Burndown Charts

# SOURCEKETTLE: PROJECTS

The project page:



The screenshot shows the SourceKettle web interface. At the top is a dark navigation bar with the 'SourceKettle' logo and links for 'Dashboard', 'Projects' (which is active), 'My Kanban', and 'Help'. On the right of the bar is a user profile for 'Tom Blount'. Below the navigation bar, the main heading is 'My Projects' followed by the subtitle 'all the projects you care about'. There are three tabs: 'My projects' (selected), 'Team projects', and 'Public projects'. A blue 'New Project' button is in the top right. The main area contains two project cards. The first card is for 'SEG\_Demo', showing 'No description available' and 'Last Modified: 48 minutes ago'. The second card is for 'ArguRDF', showing 'No description available' and 'Last Modified: on 5/3/14'. Both cards have a lock icon in the top right corner. At the bottom of the page, there are links for 'About SourceKettle v1.6.0' and 'Git help'.

SourceKettle Dashboard **Projects** My Kanban Help Tom Blount ▼

## My Projects

all the projects you care about

[My projects](#) [Team projects](#) [Public projects](#) [New Project](#)

### SEG\_Demo

No description available

Last Modified: 48 minutes ago

### ArguRDF

No description available

Last Modified: on 5/3/14

[About SourceKettle v1.6.0](#) [Git help](#)

# SOURCEKETTLE: PROJECTS

## Creating a project:


SourceKettle

Dashboard

Projects

My Kanban

Help

 Tom Blount ▾

## New Project

where baby projects are made

Name

SEG\_Demo

The 'short' name your project will be known by  
(must be a valid [UNIX](#) name)

Description

This is a demo project!

The 'long' [waffle](#) explaining your projects intent

Public

☒

Repository Type

Git

Create Project

### All SourceKettle projects include...

#### Dynamic Task Tracking

When it's more than just you working on a project, it can become troublesome keeping track of who's doing what. SourceKettle provides task/bug tracking to manage your collaborators.

#### Time Tracking

Sometimes, spreadsheets just aren't the best way of logging how much time you've spent on a task. Here's where we introduce our simple to use time logging system.

About SourceKettle v1.6.0

Git help

# SOURCEKETTLE: MILESTONES

The milestone page:

The screenshot displays the SourceKettle application interface for the 'SEG\_Demo' project. The top navigation bar includes links to Dashboard, Projects, My Kanban, and Help, along with a user profile for Tom Blount. The main header shows 'SEG\_Demo' and 'Open Milestones'. A sidebar on the left lists features like Overview, Time, Schedule, Source, Tasks, My Tasks, and Milestones (which is currently selected), as well as administration options like Collaborators, Settings, and Help. The main content area shows two milestones: 'Envisioning' and 'Sprint 1'. 'Envisioning' is active and shows a progress bar with 1 resolved (green), 1 in progress (orange), and 1 open (red) task, totaling 2/7 points. 'Sprint 1' is inactive and shows 'no tasks in this milestone' with 0/0 points. A 'Create milestone' button is located in the top right corner of the main content area.

SourceKettle Dashboard Projects My Kanban Help Tom Blount

## SEG\_Demo Open Milestones

Open milestones Closed milestones Create milestone

**Envisioning** 1 resolved 1 in progress 1 open 2/7 points

**Sprint 1** no tasks in this milestone 0/0 points

About SourceKettle v1.6.0 Git help

# SOURCEKETTLE: MILESTONES

## Creating a milestone:

SourceKettle

Dashboard

Projects

My Kanban

Help

Tom Blount

SEG\_Demo New Milestone

FEATURES

[Overview](#)

[Time](#)

[Schedule](#)

[Source](#)

[Tasks](#)

[My Tasks](#)

[Milestones](#)

ADMINISTRATION

[Collaborators](#)

[Settings](#)

[Help](#)

Short name

Sprint 2

Description

Refining of initial prototype shown to client; integration testing and prep for user studies

Start date

2015-03-02

When work on the milestone will begin

Completion date

2015-03-15

When work on the milestone will end

Submit

What is a 'Milestone'?

A Milestone (or "sprint", or "timebox") is a period of time during which a set of tasks will be completed. A Milestone can help to prevent you, and your team mates, from straying from the path of progress.

- **Step 1)** Pick a date to start work, and a target date for completion
- **Step 2)** Plan your milestone before the start date: add tasks that you want to get done
- **Step 3)** Prioritise your tasks in the "plan" view and assign work to your team
- **Step 4)** Stick to your target completion date: drop lower-priority tasks if you need to

About SourceKettle v1.6.0

Git help

# SOURCEKETTLE: TASKS

The task page:

The screenshot displays the SourceKettle application interface for the 'SEG\_Demo' project. The top navigation bar includes links for 'SourceKettle', 'Dashboard', 'Projects', 'My Kanban', and 'Help', along with a user profile for 'Tom Blount'. The main header shows 'SEG\_Demo' and 'Things to Do...'. A sidebar on the left lists various features and administration options. The main content area features a 'Task list' section with filters and a 'Create task' button. Three tasks are listed: '#1 - User Stories', '#2 - Risk Assessment', and '#3 - User Personas', each with a status icon, story points, and a user avatar.

SourceKettle Dashboard Projects My Kanban Help Tom Blount

SEG\_Demo Things to Do...

FEATURES

- Overview
- Time
- Schedule
- Source
- Tasks**
- My Tasks
- Milestones

ADMINISTRATION

- Collaborators
- Settings
- Help

Filters: (clear) Milestone Status Priority Type Assigned to Created by Create task

### Task list

#1 - User Stories R 2 SP		#2 - Risk Assessment I 2 SP		#3 - User Personas O 3 SP
-----------------------------	--	--------------------------------	--	------------------------------

About SourceKettle v1.6.0 Git help

# SOURCEKETTLE: TASKS

Creating a task:



# SEG\_Demo

 Create a task


## FEATURES

[Overview](#)[Time](#)[Schedule](#)[Source](#)[Tasks](#)[My Tasks](#)[Milestones](#)

## ADMINISTRATION

[Collaborators](#)[Settings](#)[Help](#)Subject 

50 characters max

[Save](#)Milestone  Priority Assigned to Time Estimate 

Roughly how much time will the task take to finish?

Story Points 

An abstract estimate of how complex the task is to implement

Review a bunch of software/services; decide which is best to use for the project

[Edit](#)[Preview](#)

## Issue type

- ☐ bug
- ☐ duplicate
- ☒ enhancement
- ☐ invalid
- ☐ question
- ☐ wontfix
- ☐ documentation
- ☐ meeting
- ☐ maintenance
- ☐ testing

## Subtasks/blocked by

## Unrelated tasks

## Subtask of/blocks

#1 User Stories

#2 Risk Assessment

#3 User Personas

# SOURCEKETTLE: CHARTS

Schedule:

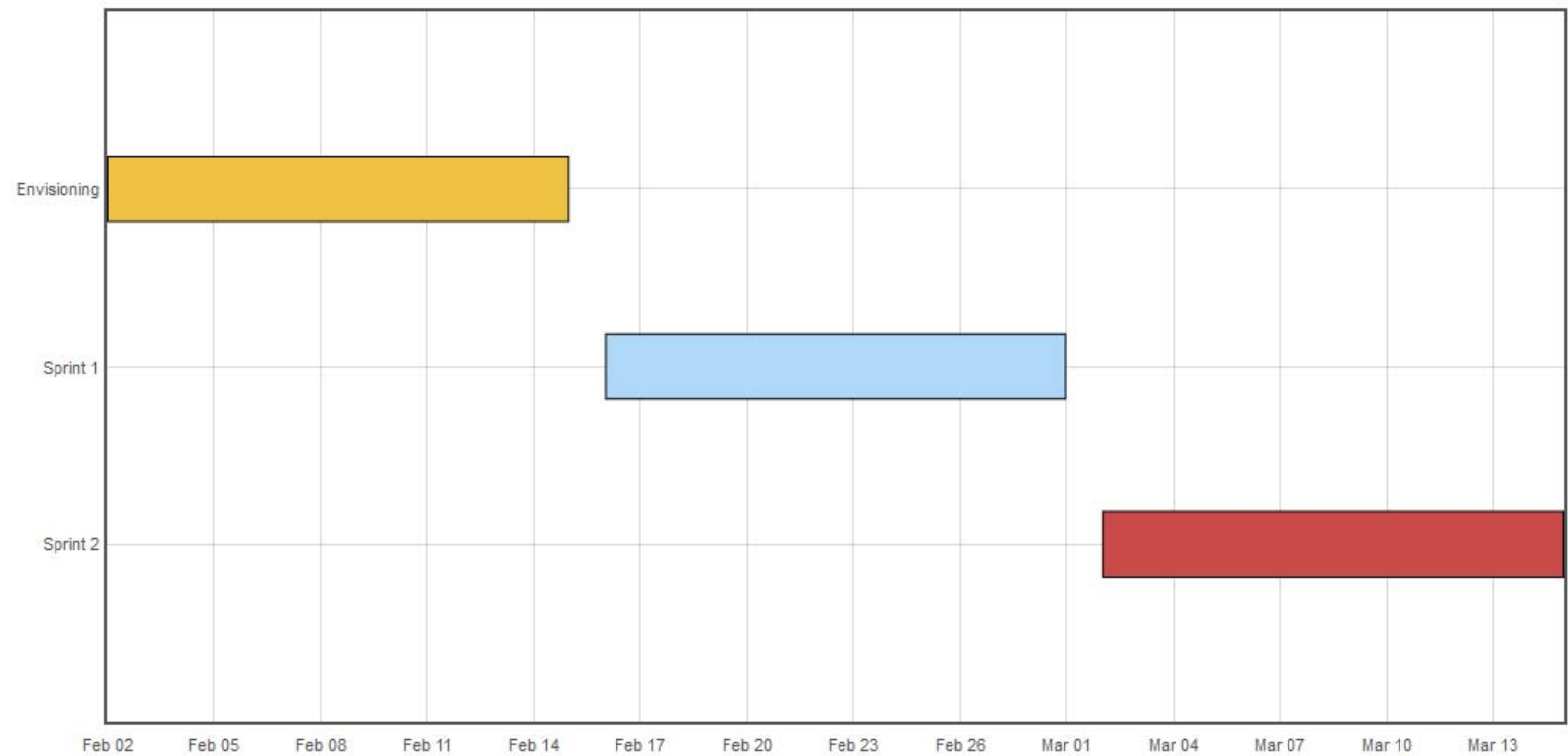
# SEG\_Demo

## Milestone schedule

## FEATURES

[🏠 Overview](#)[🕒 Time](#)[📊 Schedule](#)[🔧 Source](#)[📁 Tasks](#)[❤️ My Tasks](#)[🏠 Milestones](#)

## ADMINISTRATION

[👤 Collaborators](#)[⚙️ Settings](#)[📖 Help](#)

# SOURCEKETTLE: CHARTS

Burndown:

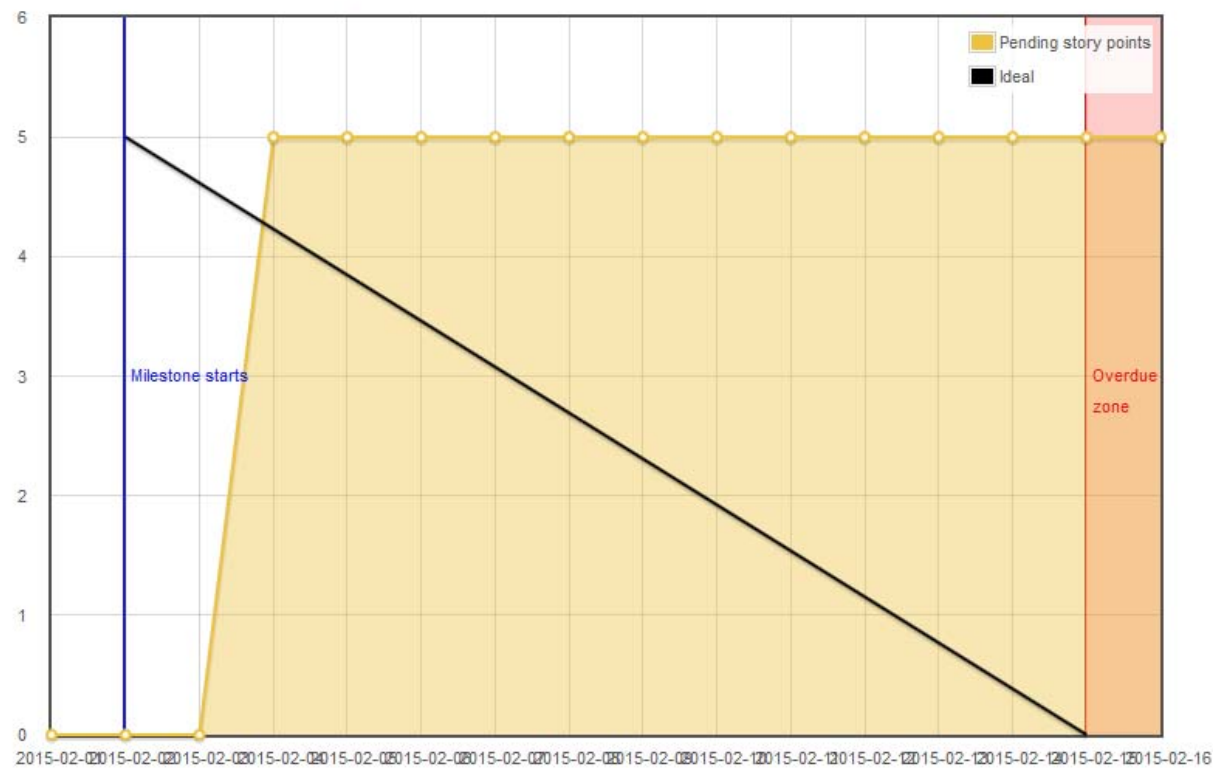
# SEG\_Demo Milestone burn-down chart

## FEATURES

- Overview
- Time
- Schedule
- Source
- Tasks
- My Tasks
- Milestones**
- Collaborators
- Help

Kanban Burn-down Priority planner Edit Close Delete

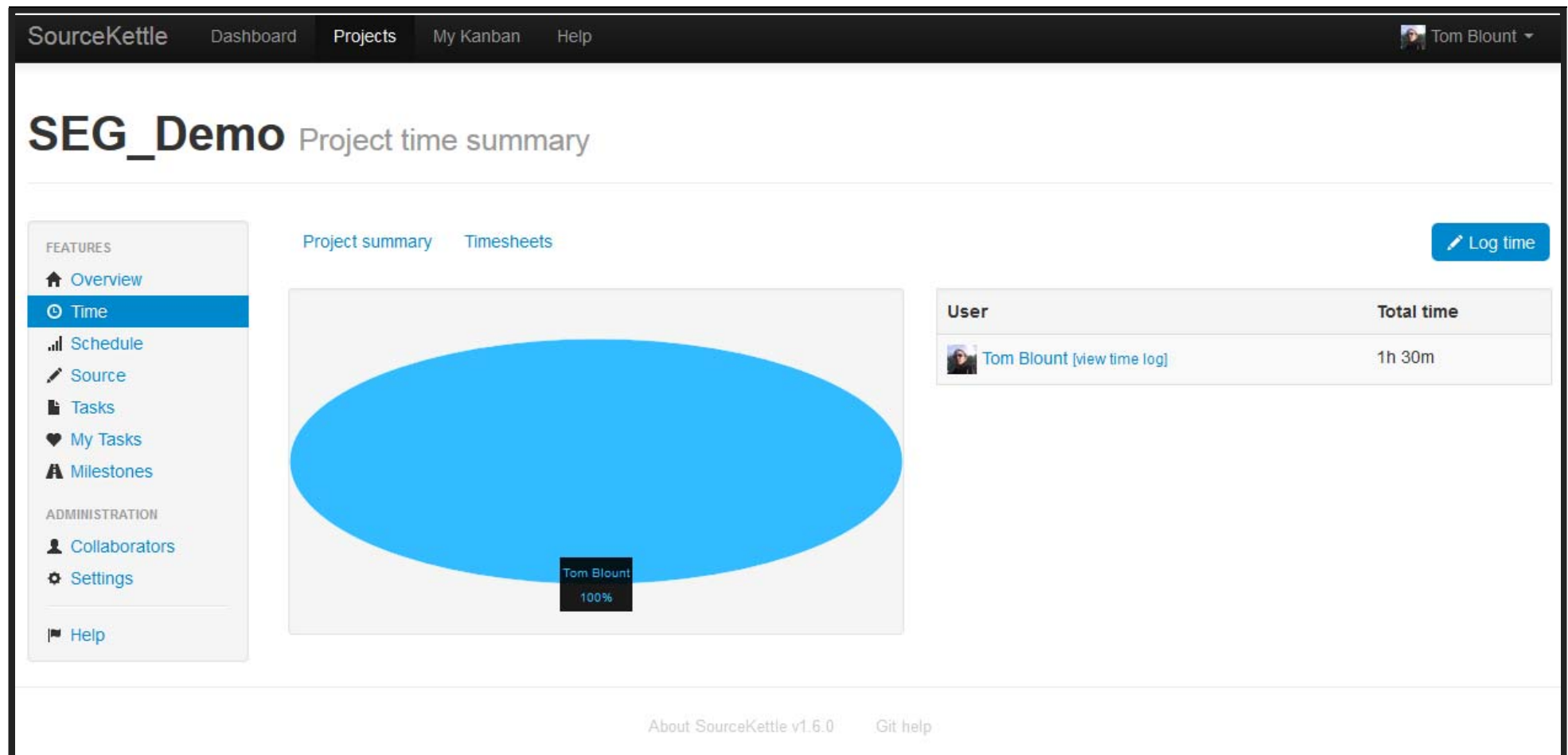
Create task



- ☒ Show story points
- ☐ Show number of tasks
- ☐ Show estimated time (hours)
- ☐ Include finished tasks

# SOURCEKETTLE: CHARTS

Time distribution:



# SOURCEKETTLE: CHARTS

## Timesheets:

SourceKettle

DashboardProjectsMy KanbanHelp

Tom Blount

SEG\_Demo

Timesheets

FEATURES

Overview

Time

Schedule

Source

Tasks

My Tasks

Milestones

ADMINISTRATION

Collaborators

Settings

Help

Project summaryTimesheets

Get CSVLog time

Task	User	Mon Feb 02	Tue Feb 03	Wed Feb 04	Thu Feb 05	Fri Feb 06	Sat Feb 07	Sun Feb 08	Total
User Stories	Tom Blount	---	---	1h 30m	---	---	---	---	1h 30m
Total		0h 0m	0h 0m	1h 30m	0h 0m	0h 0m	0h 0m	0h 0m	1h 30m

<2015-02-02 - 2015-02-08>

About SourceKettle v1.6.0

Git help

# SUMMARY

- Overview of what version control is, and why you should use it
- Brief introduction to setting up a git repo
- Setting up projects, milestones and tasks in SourceKettle
- Slides available at: <http://tomblount.co.uk/version-control>



# OTHER TUTORIALS

- [Git basics](#)
- [In-depth tutorial](#)
- [Generating SSH keys](#)

**QUESTIONS?**