

# MyMovie

Tracking movies so you don't have to.

James Mitchell



# Introduction

A series of talks covering the “life cycle” of a project.

# Talk 1

*“Louie, I think this is the beginning of a beautiful friendship.”*

Things to do before writing any code.

- Getting the requirements
- Writing down the first tasks
- Setting up tools and services
- Heavy thinking

# Pre-Production

## What is the project about?

A tool to remind me when movies I'm interested in are released...  
in the cinema, on DVD, for rental (weekly!)

# “...remind me...”

- The application needs a concept of **me**
- There is a regular process that checks stuff and reminds me

**“...movies I'm interested in...”**

The application needs a concept of **movies**, and a link from me to a movie.

**“...released to cinema, DVD, and rental...”**

A service we can query for release dates for cinema, DVD, and rental...

# Extreme Wide Shot

What should this look like?

- A responsive webapp for searching for movies and adding them to a wishlist
- A backend API for storing the wishlist and sending notifications
- A automatic deployment process, triggered by pushing code to the repository
- A development process to track work done, and yet to do

# Close Up

What are the “givens”?

- Python
- Version control system (VCS)
- Web based app in django
- Deployed into the cloud
- UI in AngularJS

# Wide shot

What additional tools are in the frame?

# Version Control

## github

Using a local version control system is a given, but also push code to a VCS in the cloud, and use API webhooks to integrate with other cloud services.

*Alternatives are Bitbucket, Gitlab, or even Launchpad.*

# Automatic Testing

## Travis-CI

Whenever changes are pushed to github, Travis-CI will checkout a copy of the code, build it, and execute tests.

The build process should also trigger a deployment to test and production servers.

*Alternatives are Jenkins or Bamboo*

# Tracking new features

## Pivotal Tracker

A tool to track new features and estimate when they can be completed.

Ideally the tool should link to the VCS and record what code implemented a feature.

*Alternatives are Jira, Trac, Basecamp, or Stick-It notes.*

# Group chat

## Slack

For a team of developers, somewhere to ask questions, hold meetings, share documents and internet memes

Also useful as a place for notifications from github and travis-ci

*Alternatives are Hipchat, Campfire, perhaps email or an XMPP client...*

# Cut to - Setting up

Let's setup the github repository, and connect it to Travis-CI, Slack, and Pivotal Tracker.

# Over the Shoulder Shot

Use a whiteboard to work stuff out

- Database
- UI flow
- API

# Noddy shot

Next time we aim to write some actual code!

Questions?