

Pipeline Monitor

Daniel Chia, Eddie Chan, Susan Jiang, James
Hohman

Work from each individual member

James - Added tornado to do server push to the pipeline, modified d3 chart to render on server push json data, added template driven responsive navbar

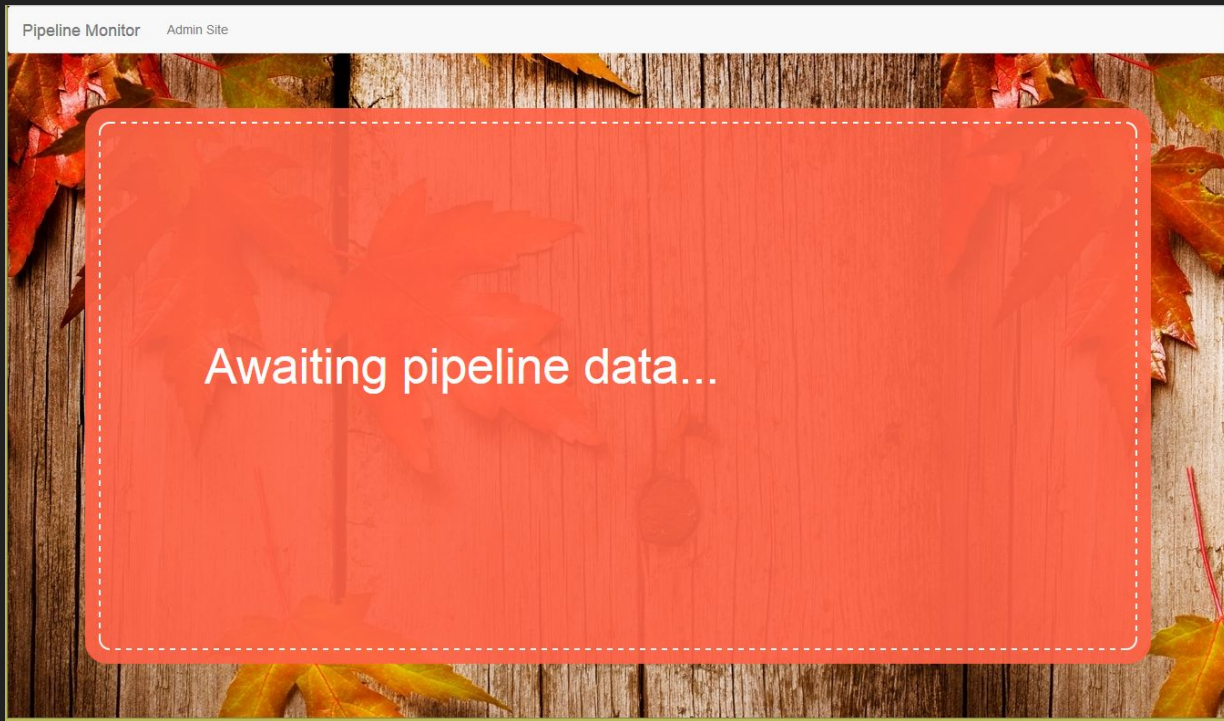
Daniel - Wrote units tests for news app, started adding selenium tests to test page features

Eddie - Fixed bugs, and added bookmark feature to news app

Susan - Used Goose library to extract article's main image and summary

Websockets

Now pipeline uses tornado server to push data to front end via websockets.



Websockets

Page is fetched from Django server and rendered.

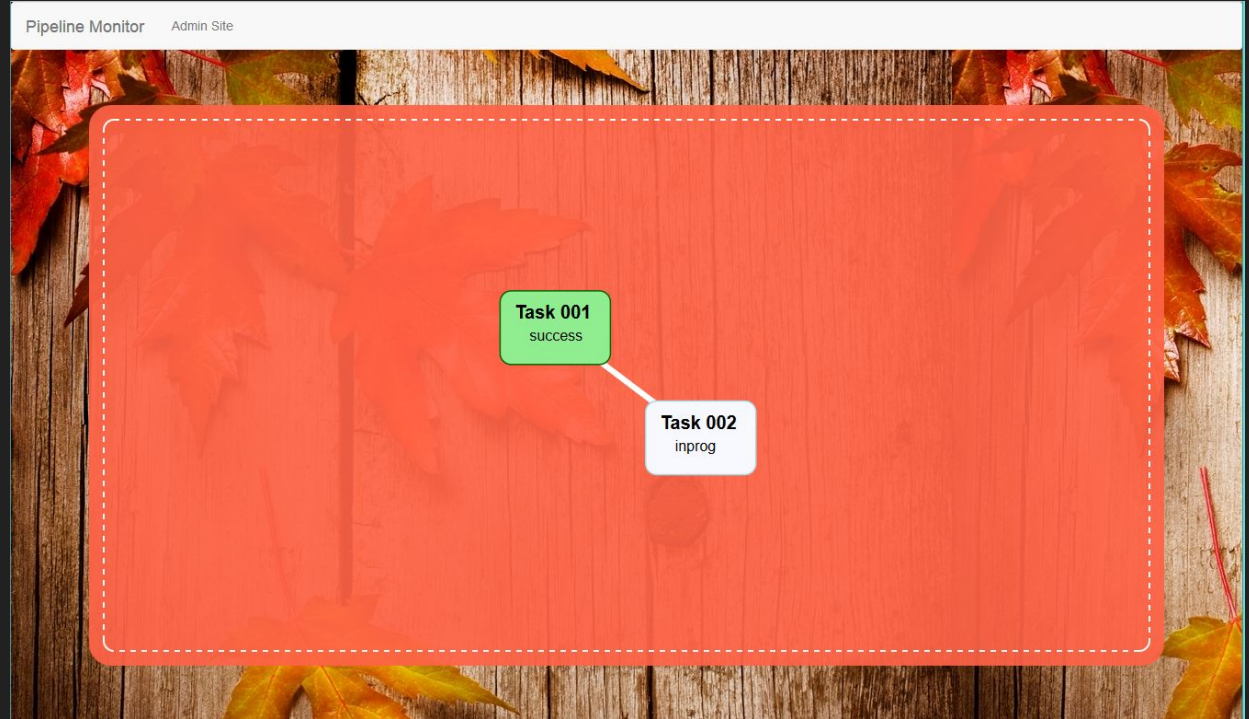
Downloads websocket.js to initialize websocket connection on load.

```
/**
 * Created by jhohman on 12/1/2015.
 */
$(document).ready(function () {
    var ws = new WebSocket("ws://localhost:8081/socket");
    ws.onmessage = function(event) {
        d3_fl_chart.render(event.data);
    };
});
```

json payload

json payload pushed
from server.

d3 force layout chart is
rendered on json
payload reception.



json payload

Here is the structure of the prototype json payload.

Will modify to include:

- Message
- Description
- URL (for detail view)

The screenshot displays the 'JSON Editor Online' interface. The top navigation bar includes 'New', 'Open', 'Save', and 'Help' buttons. The main editor area is split into two panes. The left pane shows a JSON payload with line numbers and syntax highlighting. The right pane shows a tree view of the same JSON structure.

JSON Payload (Left Pane):

```
1 [{"nodes": [{"name": "Task 001", "group": 1, "status": "success"}, {"name": "Task 002", "group": 1, "status": "inprog"}], "links": [{"source": 0, "target": 1, "value": 40}]}]
```

JSON Tree View (Right Pane):

- object {2}
 - nodes [2]
 - 0 {3}
 - name : Task 001
 - group : 1
 - status : success
 - 1 {3}
 - links [1]
 - 0 {3}
 - source : 0
 - target : 1
 - value : 40

Sphinx Documentation

Excerpt of the Sphinx documentation generated for the `pmonitor.models` class.

```
class pmonitor.models.Task(*args, **kwargs)
    Bases: django.db.models.base.Model

    Task model.

    exception DoesNotExist
        Bases: django.core.exceptions.ObjectDoesNotExist

    exception Task.MultipleObjectsReturned
        Bases: django.core.exceptions.MultipleObjectsReturned

    Task.child

    Task.child_of

    Task.get_next_task()
        Method to traverse the task hierarchy. Gets child task.

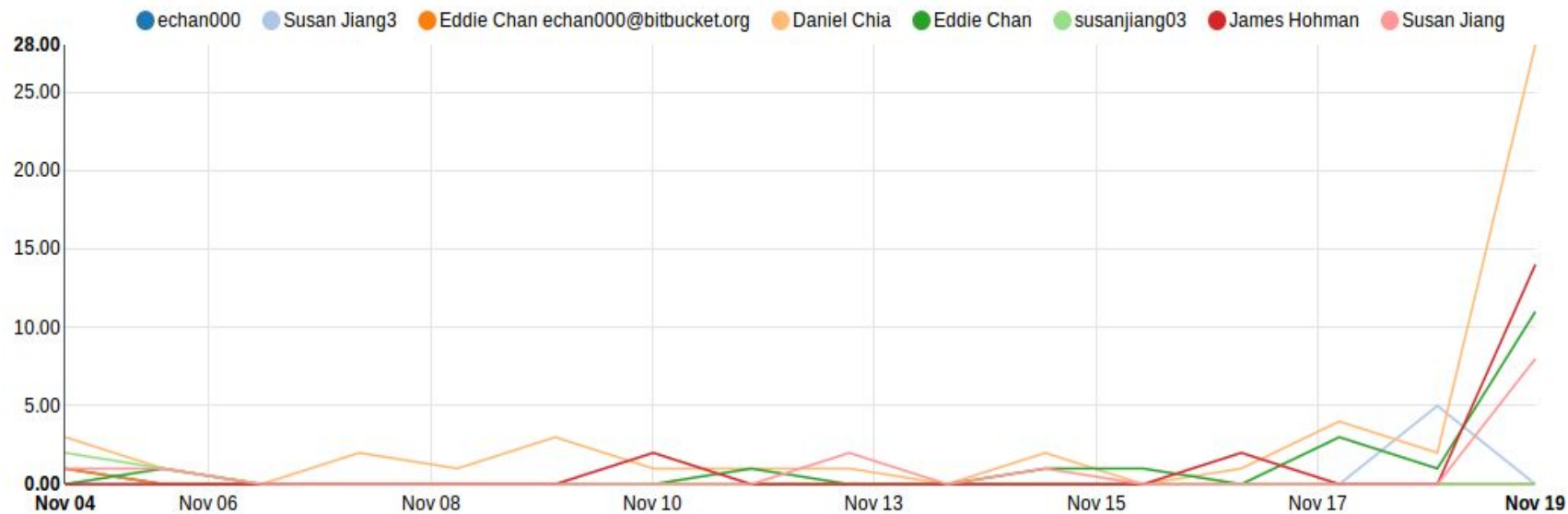
        Returns: Task
        Raises: DoesNotExist

    Task.get_previous_task()
        Method to traverse the task hierarchy. Gets parent task.
```

Technetium stats

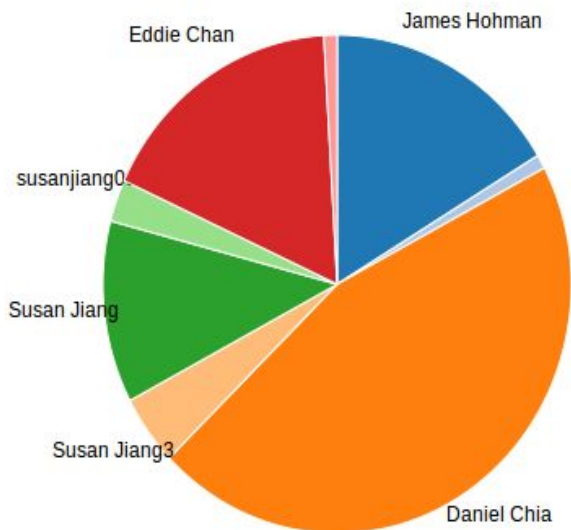
Bitbucket User	Commits	Issues Opened	Issues Assigned	Issues Completed	Issue Comments
James Hohman	18	10	15	7	9
echan000	1	0	0	0	0
Daniel Chia	50	37	13	9	38
Susan Jiang3	5	0	0	0	0
Susan Jiang	13	0	6	4	0
susanjiang03	3	0	0	0	0
Eddie Chan	19	0	12	8	0
Eddie Chan echan000@bitbucket.org	1	0	0	0	0

Commit Activity By Users

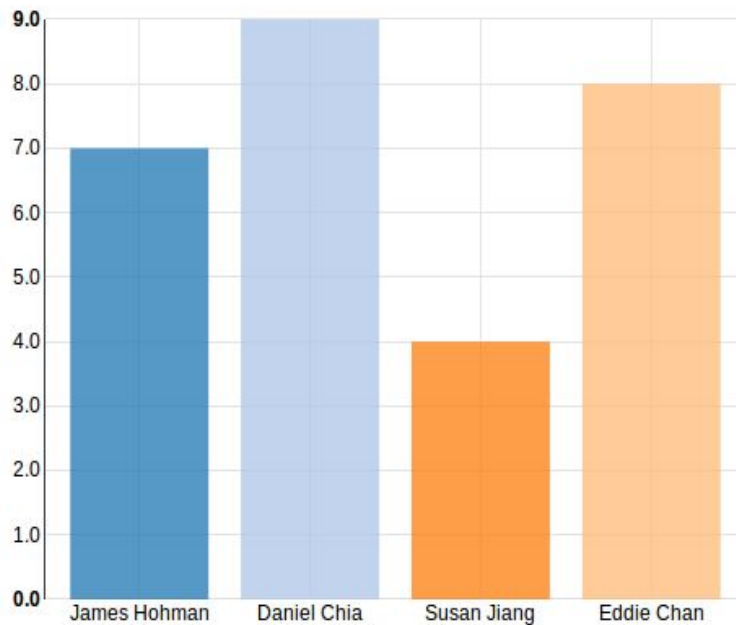


Commit Breakdown By Users

James Hohman echan000 Daniel Chia
Susan Jiang3 Susan Jiang susanjiang03
Eddie Chan Eddie Chan echan000@bitbucket.org



Issues Completed By Users



PyLint Scores on News and Pmonitor

Messages

message id	occurrences
missing-docstring	2
invalid-name	1

Global evaluation

Your code has been rated at 7.00/10

message id	occurrences
bad-whitespace	57
missing-docstring	50
no-member	38
line-too-long	26
bad-continuation	22
invalid-name	16
trailing-whitespace	15
redefined-outer-name	11
bad-indentation	7
unused-variable	5
unused-import	4
superfluous-parens	3
too-many-locals	2
pointless-string-statement	1

Global evaluation

Your code has been rated at -1.11/10 (previous run: -1.11/10, +0.00)

For the final demo day

- Bridge the tasks on the news app to the pipeline monitor
- Fix the awful pylint scores on news app
- Deploy changes to live site