Michael Jeffers (mjeffers)
Clayton Ritcher (critcher)

# Product Backlog

## Pages

Note that each of these backlog items include html/css, server code, and (possibly) javascript subitems.

### User Authentication

- Registration Page - Standard user registration page, prompting the user for a username, email, name, and password. Use email verification to confirm registration.
- Login Page - Standard login page with a form to login.

### General

- About Page - A brief explanation of the webapp that is displayed to non-logged-in users.
- Profile Page - This page allows the user to edit their profile information, import a calendar, and edit settings for their added apps. See the wireframe at the end of the document.

### Calendar/Apps

- Main Calendar Page - This page displays the user's calendar, complete with their normal calendar events (imported) and the app-generated events. Also, each app that the user has added to their profile is displayed on the side, and can be toggled on/off for the calendar. See the wireframe at the end of the document. Behind the scenes, AJAX calls are being made to each app that the user has added, asking for what to display.
- App Store Page - This page displays all of the available apps that a user can add to their account, with a brief description of each. When a user selects one to add, an AJAX call is made to that app, asking for a JSON object that describes a form that the user can fill out to specify parameters to be handed to the app on future calls (e.g. a zip code for a weather app). See the wireframe at the end of the document.

## Other

- .ics File Parser - This will be in charge of parsing the .ics calendar file and converting the events into our data model type.
- Data Model - This should include definitions for our user model, a calendar, events, apps, and 'app settings' (the user-specific instance of an app, complete with their options for that app.
- JSON to Form Converter/Protocol - A protocol and associated converter that allows apps to define a form that they want users to fill out to specify parameters for their app.

- Example Apps - We plan to create at least two apps to demonstrate the capability of our framework. Some ideas include a new movie releases tracker, weather reporter, and a sports team tracker. More ideas are listed in the proposal.

# Sprint 1

Owner: Clayton Ritcher (critcher)

## Backlog

The main goal for this sprint is to start the project, setup the basics for user authentication and profiles, and begin some of the framework backend code for interfacing with external ‘apps’.

- Data Model (Clayton)
- Registration Page (Michael)
  - HTML
  - Backend
- About Page (Michael)
  - HTML
  - Backend
- Login Page (Michael)
  - HTML
  - Backend
- JSON to Form Converter/Protocol (Clayton)

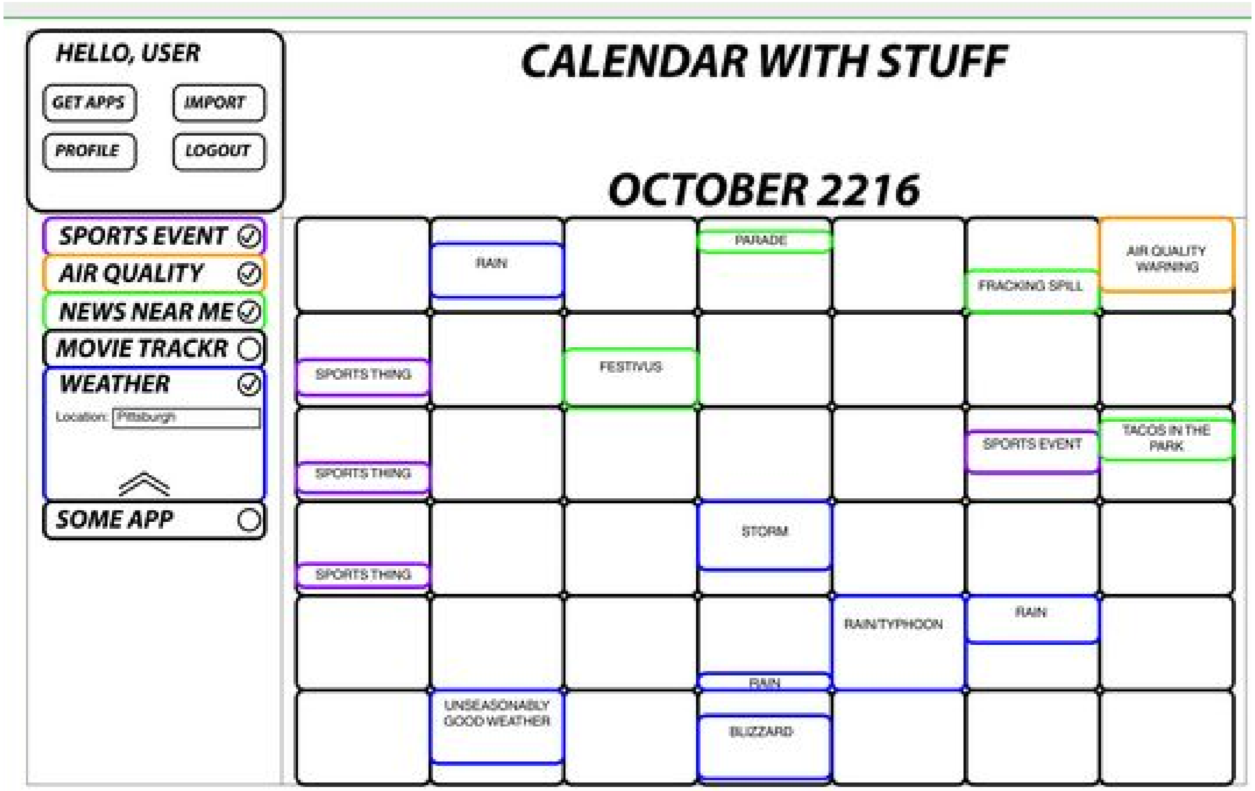
# Data Model

```
from __future__ import unicode_literals

from django.contrib.auth.models import User
from django.db import models
from django.core.validators import MaxValueValidator

class CalendarUser(models.Model):
        user = models.OneToOneField(User, related_name='userPointer')


class Calendar(models.Model):
        owner = models.OneToOneField(CalendarUser, on_delete=models.CASCADE)


class Event(models.Model):
        description = models.CharField(blank=True, max_length=100)
        name = models.CharField(blank=True, max_length=60)
        start_timestamp = models.DateTimeField()
        end_timestamp = models.DateTimeField()
        icon_url = models.CharField(blank=True, max_length=256)
        calendar = models.ForeignKey(Calendar, on_delete=models.CASCADE)
        # A handle to the AppSettings instance that created this event.
        # Blank if this event is a static calendar event.
        source = models.ForeignKey(
        'AppSettings', blank=True, on_delete=models.CASCADE)


class App(models.Model):
        description = models.CharField(blank=True, max_length=1000)
        name = models.CharField(max_length=60)
        # Can be used to detect that a user's AppSettings are out of
        # date and might need to be prompted again
        version = models.CharField(max_length=60)
        icon_url = models.CharField(blank=True, max_length=256)
        # URL that we send a GET request to in order to find out what
        # settings the user can set. Also the URL we send a POST request
        # with the user chosen settings for verification. If blank,
        # no special settings are needed
        settings_url = models.CharField(blank=True, max_length=256)
        # URL that we send the request to in order to get the actual
```

```
    # data from the app
    data_url = models.CharField(max_length=256)
    # True if the app should be allowed to be added to a calendar
    # multiple times (with different settings, maybe)
    allow_duplicates = models.BooleanField(default=True)

class Color(models.Model):
    red = models.PositiveIntegerField(validators=[MaxValueValidator(255)], default=0)
    green = models.PositiveIntegerField(validators=[MaxValueValidator(255)], default=0)
    blue = models.PositiveIntegerField(validators=[MaxValueValidator(255)], default=0)

class AppSettings(models.Model):
    settings_json = models.TextField()
    # Can be used to detect that a user's AppSettings are out of
    # date and might need to be prompted again
    version = models.CharField(max_length=60)
    app = models.ForeignKey(App, on_delete=models.CASCADE)
    # Used to let the app know when we last asked for information
    # for this user. Note that the app can still give us data
    # for dates before this timestamp (to update changed events,
    # for example)
    last_updated_timestamp = models.DateTimeField(blank=True)
    color = models.ForeignKey(Color, on_delete=models.SET_NULL, null=True,
blank=True)

    def save(self, *args, **kwargs):
    if self.color is None:
    c = Color()
    c.save()
    self.color = c
    super(AppSettings, self).save(*args, **kwargs)
```

# Wireframes

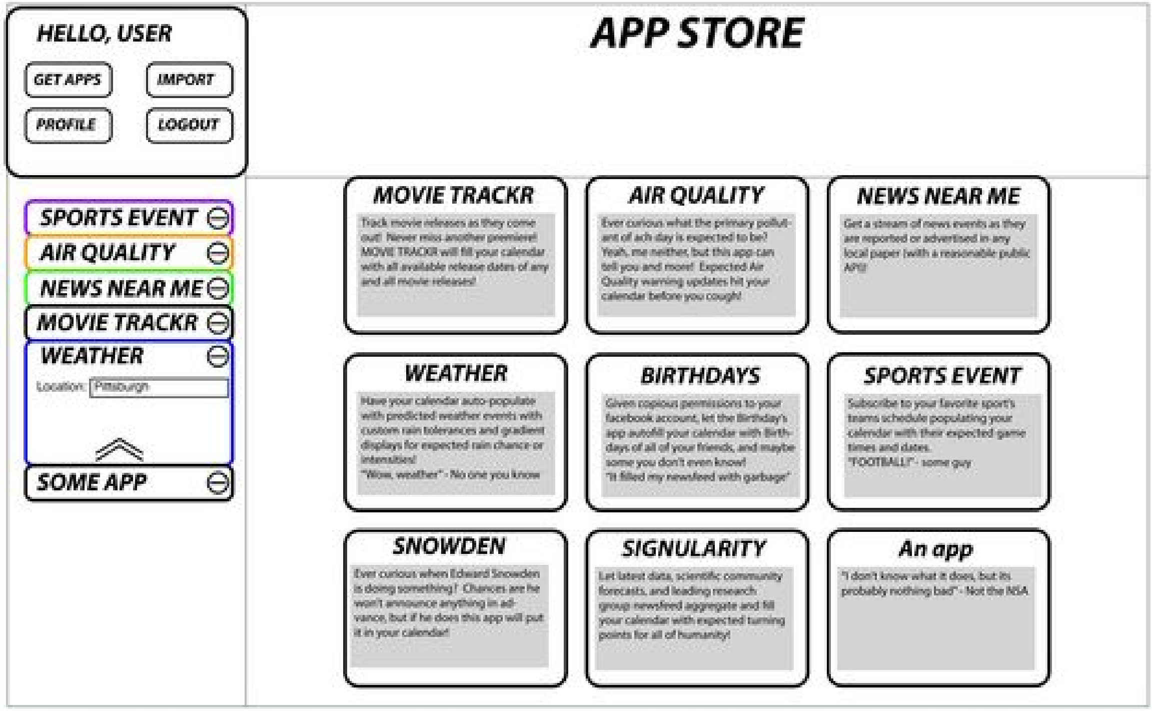
## Main Calendar Page



Location: Pittsburgh

| | | | | | | |
|---|---|---|---|---|---|---|
| | RAIN | | PARADE | | FRACKING SPILL | AIR QUALITY WARNING |
| SPORTS THING | | FESTIVUS | | | | |
| SPORTS THING | | | | | SPORTS EVENT | TACOS IN THE PARK |
| SPORTS THING | | | STORM | | | |
| | | | RAIN | RAIN/TYPHOON | RAIN | |
| | UNSEASONABLY GOOD WEATHER | | BLIZZARD | | | |

## App Store Page



Location: Pittsburgh

**MOVIE TRACKR**
Track movie releases as they come out! Never miss another premiere! MOVIE TRACKR will fill your calendar with all available release dates of any and all movie releases!

**AIR QUALITY**
Ever curious what the primary pollutant of ach day is expected to be? Yeah, me neither, but this app can tell you and more! Expected Air Quality warning updates hit your calendar before you cough!

**NEWS NEAR ME**
Get a stream of news events as they are reported or advertised in any local paper (with a reasonable public API)!

**WEATHER**
Have your calendar auto-populate with predicted weather events with custom rain tolerances and gradient displays for expected rain chance or intensities!
"Wow, weather" - No one you know

**BIRTHDAYS**
Given copious permissions to your facebook account, let the Birthday's app autofill your calendar with Birthdays of all of your friends, and maybe some you don't even know!
"It filled my newsfeed with garbage"

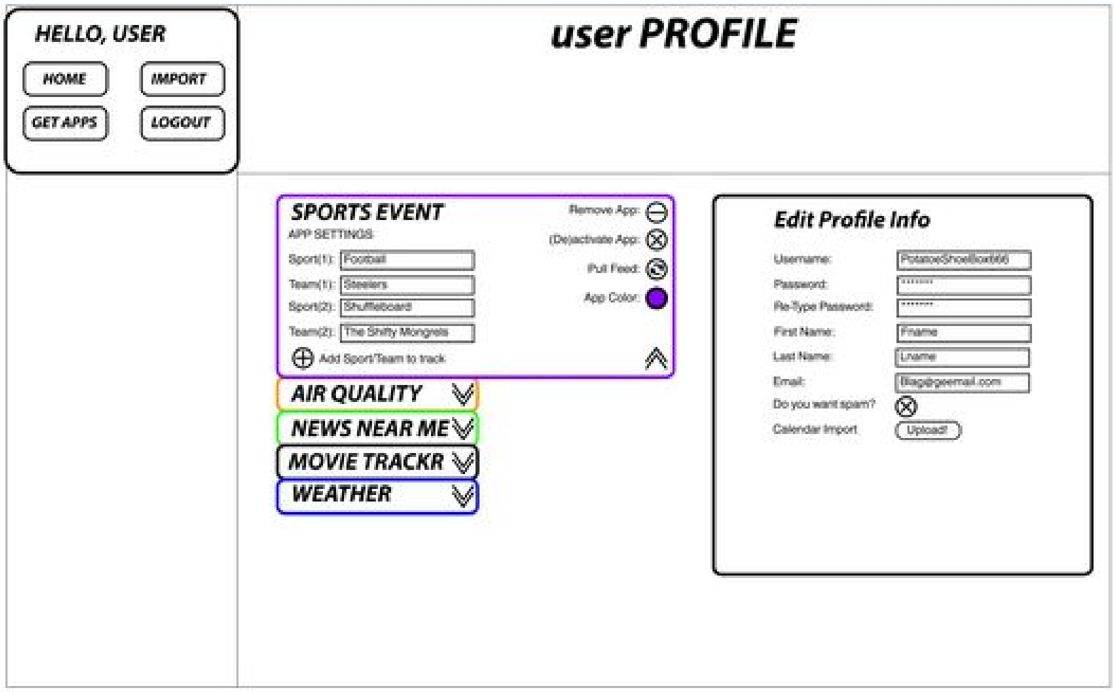
**SPORTS EVENT**
Subscribe to your favorite sport's teams schedule populating your calendar with their expected game times and dates.
"FOOTBALL!" - some guy

**SNOWDEN**
Ever curious when Edward Snowden is doing something? Chances are he won't announce anything in advance, but if he does this app will put it in your calendar!

**SIGNULARITY**
Let latest data, scientific community forecasts, and leading research group newsfeed aggregate and fill your calendar with expected turning points for all of humanity!

**An app**
"I don't know what it does, but its probably nothing bad" - Not the NSA

## User Profile Page



APP SETTINGS:

Sport(1): Football
Team(1): Steelers
Sport(2): Shuffleboard
Team(2): The Shifty Mongrels

Username: PotatoeShoeBox666
Password: ********
Re-Type Password: ********
First Name: Fname
Last Name: Lname
Email: Blag@geemail.com
Do you want spam? ⊗
Calendar Import