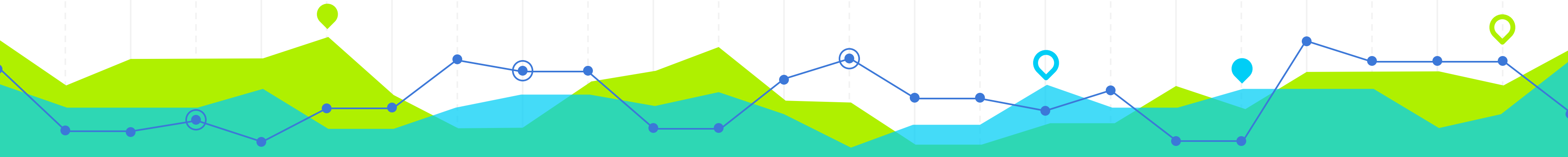
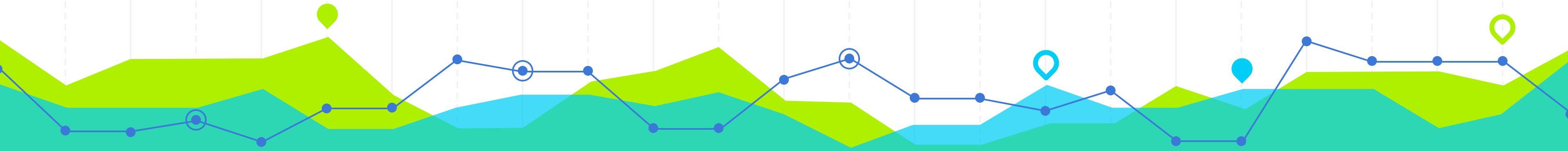


Team 201
Sprint 1

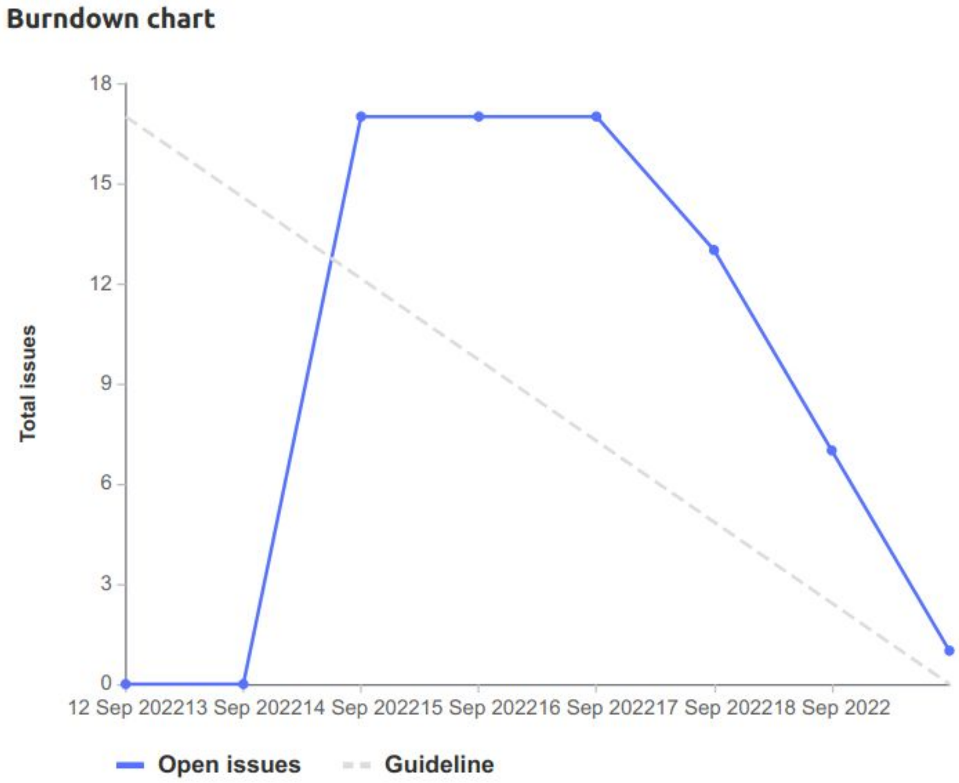


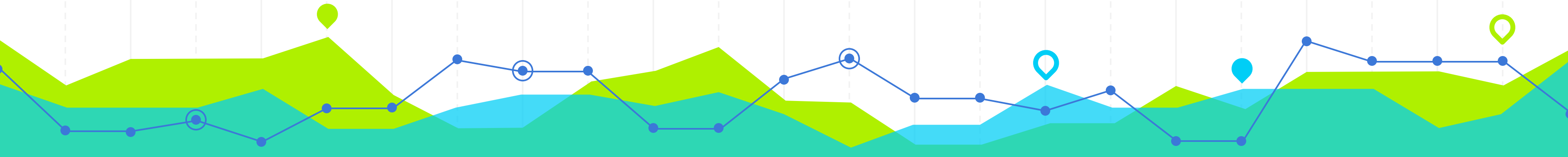
1

Burndown Chart

Let’s start with a graphical representation of our progress

BURNDOWN CHART





2

Retrospective

Let’s review what went well and how we can improve

WHAT WENT WELL AND NOT WELL

Well

- Sprint planning meeting
- Large amount of code written quickly
- Completed nearly all of our planned tasks

Not Well

- Planning happened too late
- Difficult to reach people
- Overlapping tasks caused our estimates to often be incorrect
- Waiting for people to start working on tasks

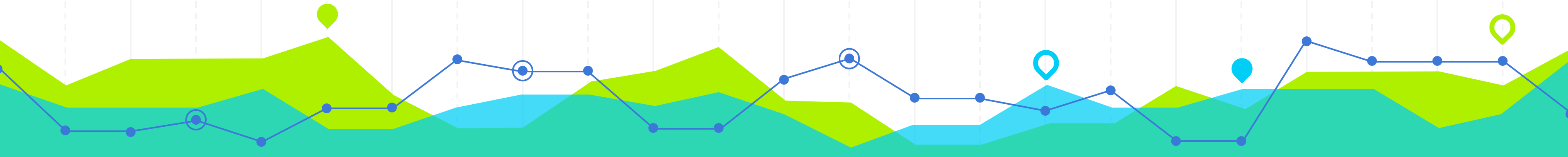
HOW WE CAN IMPROVE

Problems

- Planning happened too late
- Difficult to reach people
- Overlapping tasks caused our estimates to often be incorrect
- Waiting for people to start working on tasks

Solutions

- Start planning earlier
- Turn on notifications and be try to be reachable more often
- Spend more time planning
- Don’t assign people to tasks during planning



3

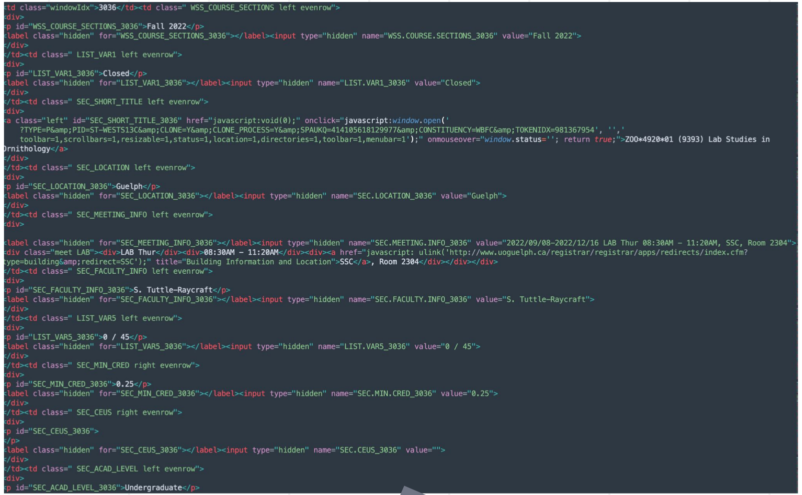
Team Member Contributions

Let’s see what we worked on this sprint

CONTRIBUTIONS - ETHAN

- Sprint 1 planning meeting (GL#19) [2h]
- Set up kanban board and created issues in GitLab (GL#8) [2h]
- Improved the project structure (GL#17) [1.5h]
- Pair programming with Sahil and Nash (GL#1, GL#2, GL#5) [4.5h]
- Added extra parsing code for our complex data structures (GL#23) [2h]
- Set up unit testing (GL#24) [1.5h]
- Add functions for displaying courses in searching program [3h]

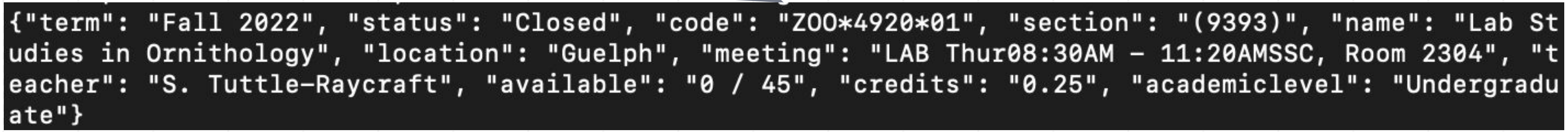
INITIAL PARSING - NASH



```
<td class="windowIdx">3036</td><td class=" WSS_COURSE_SECTIONS left evenrow">
<div>
<p id="WSS_COURSE_SECTIONS_3036">Fall 2022</p>
<label class="hidden" for="WSS_COURSE_SECTIONS_3036"></label><input type="hidden" name="WSS.COURSE.SECTIONS_3036" value="Fall 2022">
</div>
</td><td class=" LIST_VAR1 left evenrow">
<div>
<p id="LIST_VAR1_3036">Closed</p>
<label class="hidden" for="LIST_VAR1_3036"></label><input type="hidden" name="LIST.VAR1_3036" value="Closed">
</div>
</td><td class=" SEC_SHORT_TITLE left evenrow">
<div>
<a class="left" id="SEC_SHORT_TITLE_3036" href="javascript:void(0);" onclick="javascript:window.open('
   ?TYPE=P&amp;PID=ST-WESTS13C&amp;CLONE=Y&amp;CLONE_PROCESS=Y&amp;SPAUKQ=414105618129977&amp;CONSTITUENCY=WBFC&amp;TOKENIDX=981367954', '','
   toolbar=1,scrollbars=1,resizable=1,status=1,location=1,directories=1,toolbar=1,menubar=1');" onmouseover="window.status=''; return true;">ZOO*4920*01 (9393) Lab Studies in
Ornithology</a>
</div>
</td><td class=" SEC_LOCATION left evenrow">
<div>
<p id="SEC_LOCATION_3036">Guelph</p>
<label class="hidden" for="SEC_LOCATION_3036"></label><input type="hidden" name="SEC.LOCATION_3036" value="Guelph">
</div>
</td><td class=" SEC_MEETING_INFO left evenrow">
<div>

<label class="hidden" for="SEC_MEETING_INFO_3036"></label><input type="hidden" name="SEC.MEETING.INFO_3036" value="2022/09/08-2022/12/16 LAB Thur 08:30AM - 11:20AM, SSC, Room 2304">
<div class="meet LAB"><div>LAB Thur</div><div>08:30AM - 11:20AM</div><div><a href="javascript: ulink('http://www.uoguelph.ca/registrar/registrar/apps/redirects/index.cfm?
type=building&amp;redirect=SSC');" title="Building Information and Location">SSC</a>, Room 2304</div></div></div>
</td><td class=" SEC_FACULTY_INFO left evenrow">
<div>
<p id="SEC_FACULTY_INFO_3036">S. Tuttle-Raycraft</p>
<label class="hidden" for="SEC_FACULTY_INFO_3036"></label><input type="hidden" name="SEC.FACULTY.INFO_3036" value="S. Tuttle-Raycraft">
</div>
</td><td class=" LIST_VAR5 left evenrow">
<div>
<p id="LIST_VAR5_3036">0 / 45</p>
<label class="hidden" for="LIST_VAR5_3036"></label><input type="hidden" name="LIST.VAR5_3036" value="0 / 45">
</div>
</td><td class=" SEC_MIN_CRED right evenrow">
<div>
<p id="SEC_MIN_CRED_3036">0.25</p>
<label class="hidden" for="SEC_MIN_CRED_3036"></label><input type="hidden" name="SEC.MIN.CRED_3036" value="0.25">
</div>
</td><td class=" SEC_CEUS right evenrow">
<div>
<p id="SEC_CEUS_3036">
</p>
<label class="hidden" for="SEC_CEUS_3036"></label><input type="hidden" name="SEC.CEUS_3036" value="">
</div>
</td><td class=" SEC_ACAD_LEVEL left evenrow">
<div>
<p id="SEC_ACAD_LEVEL_3036">Undergraduate</p>
```

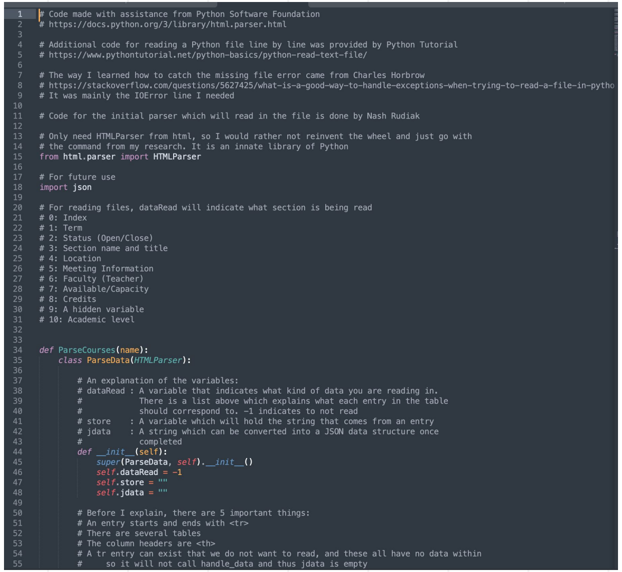
- Need to translate into something we can use
- To do this we need to read each tag and content, determine it is useful, and turn it into a string which we can convert into a JSON object
- Once read it should be easy to store



```
{"term": "Fall 2022", "status": "Closed", "code": "ZOO*4920*01", "section": "(9393)", "name": "Lab St
udies in Ornithology", "location": "Guelph", "meeting": "LAB Thur08:30AM - 11:20AMSSC, Room 2304", "t
eacher": "S. Tuttle-Raycraft", "available": "0 / 45", "credits": "0.25", "academiclevel": "Undergradu
ate"}
```

INITIAL PARSING - NASH

- Did research for understanding the HTML document
- Created code to grab content, ignoring HTML elements that are irrelevant for the data set
- Created code to turn content into a JSON string
- Documentation for courseparser.py
- Additional help for improving code was done by Ethan and Sahil



```python
# Code made with assistance from Python Software Foundation
# https://docs.python.org/3/library/html.parser.html

# Additional code for reading a Python file line by line was provided by Python Tutorial
# https://www.pythontutorial.net/python-basics/python-read-text-file/

# The way I learned how to catch the missing file error came from Charles Horbrow
# https://stackoverflow.com/questions/5627425/what-is-a-good-way-to-handle-exceptions-when-trying-to-read-a-file-in-pytho
# It was mainly the IOError line I needed

# Code for the initial parser which will read in the file is done by Nash Rudiak

# Only need HTMLParser from html, so I would rather not reinvent the wheel and just go with
# the command from my research. It is an innate library of Python
from html.parser import HTMLParser

# For future use
import json

# For reading files, dataRead will indicate what section is being read
# 0: Index
# 1: Term
# 2: Status (Open/Close)
# 3: Section name and title
# 4: Location
# 5: Meeting Information
# 6: Faculty (Teacher)
# 7: Available/Capacity
# 8: Credits
# 9: A hidden variable
# 10: Academic level


def ParseCourses(name):
    class ParseData(HTMLParser):

        # An explanation of the variables:
        # dataRead : A variable that indicates what kind of data you are reading in.
        #            There is a list above which explains what each entry in the table
        #            should correspond to. -1 indicates to not read
        # store    : A variable which will hold the string that comes from an entry
        # jdata    : A string which can be converted into a JSON data structure once
        #            completed
        def __init__(self):
            super(ParseData, self).__init__()
            self.dataRead = -1
            self.store = ""
            self.jdata = ""

        # Before I explain, there are 5 important things:
        # An entry starts and ends with <tr>
        # There are several tables
        # The column headers are <th>
        # A tr entry can exist that we do not want to read, and these all have no data within
        #     so it will not call handle_data and thus jdata is empty
```

CONTRIBUTIONS - AMEER

- Developed initial searching, and worked with Kal to complete
- Developed unit tests for CourseParser with Sahil
- Wrote documentation on our JSON format
- Researched docker implementation with Python

CONTRIBUTIONS - SAHIL

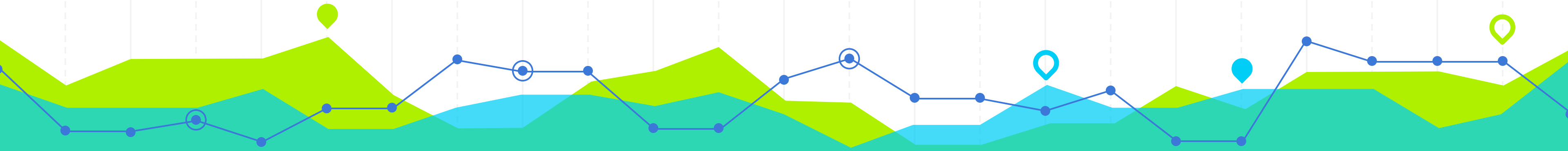
- Sprint 1 planning meeting (GL#19) [2h]
- Improved the project structure with Ethan (GL#8) [1h]
- General programming with Ethan [2h]
- Designed course data structures(GL#2) [1h]
- Researched and documented the format of the exported data (GL#3) [1h]
- Exported parsed data into a file (GL#5) [2h]
- Wrote Unit tests for untested code (GL#10) [2h]

CONTRIBUTIONS - DOGU

- Tested the code on different platforms Windows, Linux, Mac.
- Sprint1 planning meeting and code review session.
- Researched VBA.

CONTRIBUTIONS - KALINDU

- Developed initial searching by course code with Ameer
- Developed advanced course searching:
  - Searching by course name
  - Searching by professor name



4 Demo

A live demonstration of our work so far

CREDITS

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash