Real Time Wine Sensing Tool

PROCESS PRESENTATION

Scrum

Scrum Masters

- Harnoor, Harrison, Ross, Jay and Vandit

Effectiveness of Scrum Meetings

- Meetings themselves were generally efficient

Scrum procedure in the project

- Scrum master led the meetings
- Twice a week
- Would include an agenda

Effectiveness of team communication

- Definitely room for improvement here

Requirements

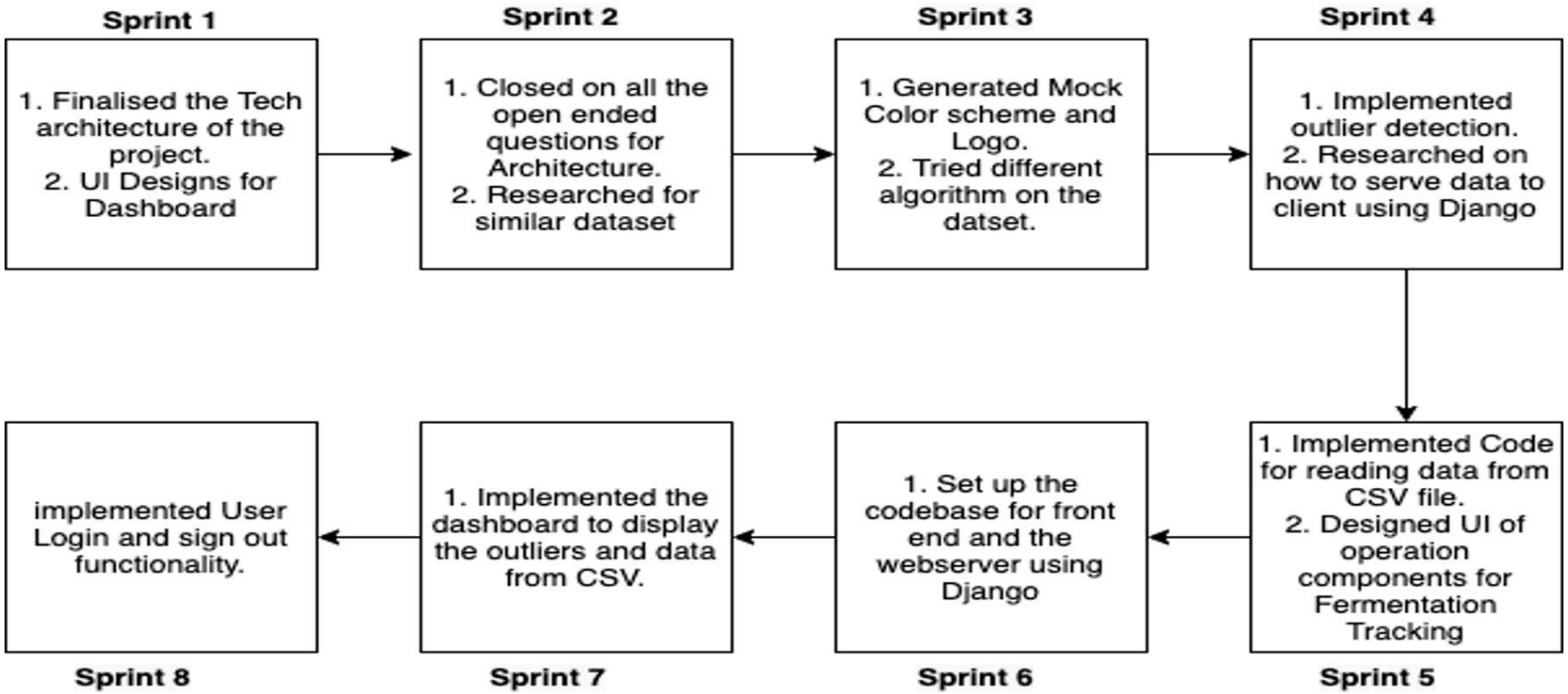
- The team was provided with initial skeletal requirements for the project from the client.
- Additional requirements were gathered at the beginning of fortnightly sprints with meetings with the client/tutor. These meetings consisted of demonstrating the software as well as asking questions to set up a new sprint of more-specific tasks.
- An early hurdle was introduced with the unavailability of the wine data necessary to build the application's features around. This was mitigated by finding some similar publicly-available data to temporarily take its place.
- Requirements were then changed later in the project with the late arrival of the wine data:
  - Some major data-dependent requirements had their expectations adjusted to match the little remaining time.
  - The changes also allowed for some additional user-authentication requirements to be scheduled.
- In terms of the impact on our team's progress, we were able to switch to the change in requirements without it hindering our overall progress.

Definition of Done

DOD AT THE START OF THE PROJECT

- Research on the Tech Architecture.
- Finalize the tech stack for Infra.
- Finalize the tech stack end to end.
- UI Design for Dashboard.
- Implement Front end for dashboard.

HIGH LEVEL EVOLUTION OF DOD



Definition of Done

DOD AT THE END OF THE PROJECT

- The sprint was planned and mutually understood, reviewed, and accepted by all members of the team.
- The current snapshot (Described in the ‘User Stories’) have met respective acceptance criteria for individual items.
- CSV Loader and Outlier detection module have been implemented.
- Designing of the UI and merge the outlier detection module with the dashboard in the final stage for completion.
- The implementation for the Dashboard and Sign-in options are currently in working stage.

Definition of Done

Reasons for Evolution

- Proactiveness of the team.
- Sprint goals.
- Dataset.
- Feedback from the clients.

Tools - Communication

FACEBOOK MESSENGER

- A project group in Facebook Messenger was our primary means of communication
- Familiarity with the platform made it an easy first choice

DISCORD

- Discord was used for more official communication
- Channels for discussing with tutor and sharing progress

ZOOM

- Our weekly meetings were held via Zoom

FUTURE – SLACK

Tools – Planning & Task Management

GITHUB PROJECT BOARDS

- We assigned tasks/planned sprints using Github's Kanban board
- Extremely useful as it localized code and tasks into one place
- While very useable, it also has very limited functionality

FUTURE – JIRA

- Many popular alternatives are available, however most cost money which we did not have
- If taking this project further perhaps use something like Jira

Tools - Design

UI BAKERY

- UI Designs were initially made in UI Bakery, a visual web app builder
- Ease of use and quality of results made this an ideal starting point

ADOBE PHOTOSHOP

- More nuanced and customizable designs
- Images produces used to guide HTML/CSS for website

Testing

How we tested the software

- Creating tests as major parts were completed

Pushing bugs to Github

- Tested offline before being pushed
- Offline testing safeguarded

Use of testing frameworks

- Testing frameworks were not used

Testing

Use of Test-Driven Development

- Test-driven development not used
- There may have been benefits to using it
- Would Test-Driven Development work with our process?

Adequacy of testing

- How did our process adequately test our program?

Progress Analytics

- **Early Roadblocks**
  - Delays with receiving our required dataset
    - Progressed with alternate dataset
  - Initial group organisation issues
    - Eventually began to organise regular meetings
- **Sprint 1**
  - After initial roadblocks, we made good progress
    - Mainly planning and research tasks
    - Finished 12 points

Progress Analytics

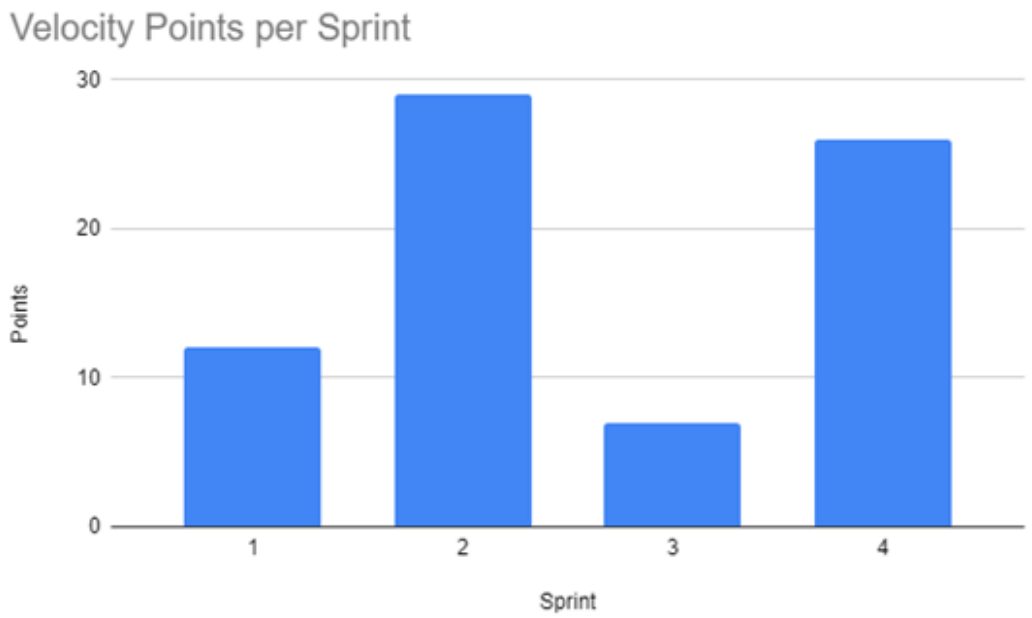
- **Sprint 2**
  - This was our most productive sprint points-wise
    - Implemented ML algorithms and completed lots of research
    - Finished 29 points worth of tasks
- **Sprint 3**
  - Many issues this sprint, including low drive due to the holidays
    - Completed some UI tasks
    - Finished only 7 points worth of tasks

Progress Analytics

- **Sprint 4**
  - Very productive sprint
    - Implemented many modules for the front end and back end
    - Finished 26 points worth of tasks
- **Overall**
  - Our process was mostly sound
  - Possible improvements:
    - Begin planning earlier, to avoid inefficient start
    - Streamline tasks to prepare for upcoming sprints better
    - Plan around holidays, to keep morale up for breaks

Progress Analytics

- **Sprint Progress Visualized**



Documentations

Our group has created documents which are listed below:

- Tech-stack specifications
- Application tests
- Outlier Detection and CSV module readme files

Documentation Readers

The documents our group had created were meant for the following readers:

- Tech-stack specifications – **Application Developers**
- Application tests – **Application Testing Team**
- Outlier Detection and CSV module readme-files – **Maintenance Team**

Procedures/Standards For Documentation

The following are the steps the team has adopted for the documentation.

- Created initial document outline
- Prepared the first draft
- Conducted team review
- Completed the first draft
- A conducted independent review (Machine Learning Industrial Scientist)
- Redrafted the document based on the generous feedback
- Finalized the document

Procedures/Standards For Documentation

Reasons to follow the process:

- To ensure the document is technically accurate and is free of error.
- By following the process it improved the team's operational agility by gleaning insight from mistakes and successes.

Group Management For The Documentation

The following are the aspects where the team did a remarkable job in the documentation process.

- Timely completion of their individual assigned task.
- Provide constructive feedback to other team members.
- Reflect on the response given by the industrial expert.

List of Documents Which Were Needed

Following are the documents that our team believed would have enhanced the project implementation procedure:

- Risk mitigation plans
- Quality control and management
- Document to store committed changes in the project
- Reference document of system functionality for wine-makers

Software Analytics

- **Circumstances that lead to better code**
  - Keeping the same people working on similar areas
- **Analytics**
  - Track the sidebar navigation to see how often users are switching between tabs
  - Could some of the features be combined onto the same screen to save clicks? - Perhaps some are not being used
  - How large are the datasets our users are frequently using – is it is getting large enough to not comfortably fit on the screen?

Final Reflection - Strengths

- **Team collaboration**
  - **Example**: One of our team members was given a task to develop a user dashboard. He was facing issues implementing it, so the team collaborated to resolve the issues.
  - **Benefits**: Timely completion of the task, Efficient implementation, and Exchange of knowledge.
- **Equal distribution of workload**
  - **Example**: One of our user-stories was to design a machine-learning model which was a heavy task for one member therefore the team agreed to divide the load amongst 3 members.
  - **Benefits**: Reduced stress on a single team member, Impartial distribution of the task.

Final Reflection - Strengths

- **Adaptation to change**
  - **Example**: During the initial phase of the project due to legal issues the team did not have had received the dataset, however, the team progressed with the model with a similar open-sourced dataset.
  - **Benefit**: Overcoming barriers and progressing in an uncertain agile environment

Final Reflection - Weaknesses

- **Meeting management**
  - **Example**: It was difficult to accommodate team members' time preferences for scrum meetings because of work commitments.
  - **Effect**: Some of the members were not able to join all the meetings which caused communication gaps that led to delays in implementation.
  - **Improvement**: Prepare meeting minutes and circulate with the team members and obtain feedback to ensure correct information is received by the absent team member.
- **Improper planning for the final execution**
  - **Example**: The final integration was not completed as team members were busy with other assignment workloads.
  - **Effect**: We were not able to implement the final integration task.
  - **Improvement**: Effective time management and task planning by individual team members.

Final Reflection - Weaknesses

- **Incomplete documentation**
  - **Example**: We were not able to create a Risk mitigation plan, Quality assurance, and Changelogs during the project timeframe.
  - **Effect**: Lack of references for the application developer and maintenance team.
  - **Improvement**: Creating these documents would have helped the maintenance team to carry out system modification and enhancements after system handover.

Questions

Thank You