## Requirement

- "There is a variety of test data covering the United Kingdom, which the students should be able to use to show the percentage coverage of a geographic area covered by products held in SCi-Discover and visualise the GeoJSON associated with these products on a map. As a stretch target, it'd be good to visualise how this coverage built up over time in a histogram.
- Initial spec from slides -> created our own simple spec and sent to Claire in which she said it was accurate
- We have plotted the geojson and worked out the area of the polygon, uk and created a percentage
- We have also shown this data through multiple histograms, one of these shows the build up over time.
  EXTRA FUNCTIONALITY
- All our extras kept in mind the focus of the project, to visual data.
- The search bar helped this by allowing certain data to be shown
- The histograms updated with this information giving more in-depth analysis of the data
- Filtering also enabled this

## Complexity

- ?

## Client Focus

- Multiple emails to and from the client
- Each time refined our spec and showed our progress
- Once the feedback that was given was completed, we emailed back the result letting them know on the overall progress
- Added features that wanted as seen in requirements

## Quality and Testing

- Black box and white box testing
- Black box: gave our features to other members of the group to sit through and bug check. They knew the intended use.
- White box: tested our own features following our test scripts
- Show test table

## Group Work

- Show gitub/trello and give user anecdotes

## SCi-Toolset talking points

- Search bar
- Filtering
- Mission search/polygon search
- Area calculated for each polygon
- Percentage calculated for each polygon
- Rough estimate of missions in each county using geojson data from the government
- Histograms showing extensive data analysis and working with the search bar
- Cropping the data so only the data over the UK is shown

How to improve?

- Performance, next time we will do processing on the server side instead of client so we can utilise more cores and not freeze the browser of the client while it is loading.
- We could also use a different library
- Utilise the github
- Use d3.js instead of poltly to allow for more extensive data analysis (colours etc)