Pre-planning

# Sprint 3

Goals:

* **The project is in a demoable state (whatever current functionality is) and can be tested by TPE (threat protection engineering) team. Repo is ready to be shared and get feedback from F-Secure. (UI, API, interaction, scraping)**
  + Ability to pass custom parameters to the scrapper from the UI.
* **Processing of scraped data (fake/non-fake shops) to be used to train the model. (ML)** all stages before training should be done, integrated with existing extractor.

# Sprint 4

**Goals:**

* UI:
  + Improvements on the UI (based on feedback of the Demo)
* API:
* Scraping/Crawling:
  + Optimising performance of the scraper. Technical comparison of scraping components (pycurl, aiohttp, python native requests) & suggestions for utilising the most of the computational/network resources without degrading performance
* Extraction:
  + Major focus on feature extraction - what currently exists; what could be added based on thesis
  + Towards ML: feature summaries, distributions, statistics and graphs etc as based on work in thesis.
* General:
  + Improvements in documentation

**Tasks:**

* Review of currently existing extraction component
  + How long does extraction take for a single site? What elements of the extractor take the longest time?
  + Text based features are currently slow - can this be improved?
* Determining which features that are discussed in the thesis are already present in the output of the extractor

# Sprint 5

* Note that the second project review will happen within sprint 5 on Tuesday 5 March

**Goals:**

* Preparation for Project Review 2: update Process Overview, DoD, create slides, demo, software quality (Most important quality attributes, Quality metric)
* Start peer testing?
* UI:
  + Improvements on the UI (based on feedback): Implement the Result page
* Scraping/Crawling:
  + Auto-healing, error handling for Scraper
  + Crawler work (optional)
* Integrate ML model into the whole solution.

# Sprint 6

* Make sure the project can be easily installable (\*\*\*)
  + One possible solution to problems with requirements etc is to setup a Docker image for installation of the project
* Student peer team feedback session
  + Use one of the burner computers for the peer team to test.
  + One session with the peer team to test each other's application.
* UI: Improvements (needs project to be installable)
* ML: Improvements (needs project to be installable)
* Working towards project to be handed-over/delivered
  + DoD/acceptance criteria/documentation…
  + Example: working towards merging all the branches; doing a review of all the necessary libraries (pyspark not necessary) etc

# 

# Sprint 7

* Documentation and hand-over is the main focus for the entire project
* Working towards project to be handed-over/delivered
  + merging all the branches
  + Delete unneeded branches
  + Preparation for Project Review
  + Preparation for Gala (poster, stand)
* ML: Improvements
  + Focus on documentation and cleaning the structure
  + Provide the models
  + A summary of the data/requests/…etc
  + Give the data (at least summary of whether could scrape those 1000s of sites)

**Additional:**

* Fixing any issues that are found (e.g., CLI issue mentioned)
* Docs for Docker based development (ie cp’ing local code into container for development)
* UI: Improvements
  + When there is an error, display it
  + Feedback given:
    - Functionality
    - Loading icon
    - …

Overflow:

* Extractor:
  + Comments extracted?
  + Cookies extracted after loading?
* Auto-consent + adblocker (UI + backend)
* Event based logs - order of execution of js events for example (playwright)
* Pre-scraping task for interactivity use case: some websites allow for injection of js on the website (playwright).