**Project Requirements - OSINT Dashboard  
William Campbell, Andrew Flora, Santiago Colla**

**Business Requirements**

* Share insightful information about webpages, media, and personal identifiable information, which may be used in various OSINT investigations or for personal research.
* Help with OSINT investigations by utilizing a collection of validated technologies within one interface.
* Sharing descriptive information about various technologies, which may help users further understand specific technologies.
* Provide users with information from uploaded files, such as metadata and hash verification.
* Provide users with information regarding searched personal identification information.

**User Requirements**

* Users are able to input various types of information, including website URLs, files, and personal information.
* Users are able to obtain an automated downloadable report for the inputted URL, file, or personal information
* Users are able to scroll through generated content.
* Users are able to access the web tool via any web-browser in both desktop and mobile devices.
* Users are able to access additional information on the tools and their findings through the user interface.

**Functional**

* Website URL analysis including IP, server, domain, website features, DNS records, host names, and various URL checks.
* File analysis to receive metadata and hash verification.
* Phone number reverse lookup to receive carrier and user information.
* Email reverse lookup to receive name or identification information.
* The application will be able to access information via APIs.

**Non-Functional**

* Error checking for invalid URLs or unavailable hosts.
* Fault tolerance with automatic error detection.
* Deliver IP/DNS/Server information in under 20 seconds.
* The system must be available 24/7.
* The system must be HTTPS secure.
* The system must tolerate a load of at least 1000 simultaneous requests.

**Implementation**

* Prioritization of functional requirements over non-functional ones and cosmetic features throughout the system initial development.
* API integration for data retrieval and analysis from multiple sources.
* Information should be implemented into a custom generated report.
* Efficient methods to parse and process raw data into standardized reports.
* Tool should be accessible from a majority of web-browsers.

|  |  |  |
| --- | --- | --- |
| Use Case Name:URL Input | ID: 1 | Importance: High |

|  |  |
| --- | --- |
| Primary Actor: User | Use Case Type: Essential |

|  |
| --- |
| Stakeholders:  **Users**: Begin URL analysis process by providing a valid URL as input to the web tool. |
| Brief Description: This use case describes the process in which the user can provide valid and publicly discoverable URLs as input for analysis and processing. |
| Trigger: User inputs URL via a forum on the web page Type: External / Temporal |
| Relationships:  Associated with User Input, Report Generation, and Sitemap Parser |
| Normal Flow of Events:   1. The user accesses the web tool through a web browser on their desktop or mobile device. 2. The user selects the option to input a valid URL for analysis within the web tool's interface. 3. The web tool prompts the user to upload the desired URL. 4. The user types or pastes the URL they want to be analyzed. 5. The web tool receives the uploaded URL and begins processing the web data for the report generation. |
| Subflows:  The web tool sends the HTTP requests necessary to compile the data and formats the results |
| Alternate Flows:  S-5: If the uploaded URL is not valid or belongs to an unavailable domain, an error message will be displayed. |

|  |  |  |
| --- | --- | --- |
| Use Case Name: File Input | ID: 2 | Importance: High |

|  |  |
| --- | --- |
| Primary Actor: User | Use Case Type: Essential |

|  |
| --- |
| Stakeholders:  **Users**:Begin file analysis process by providing specific files as input to the website |
| Brief Description: This use case describes the process in which the user can provide specific files as input for analysis and processing |
| Trigger: Input a valid file using an upload forum on the webpage. Type: External / Temporal |
| Relationships:  Associated with User Input and Report Generation |
| Normal Flow of Events:   1. The user accesses the web tool through a web browser on their desktop or mobile device. 2. The user selects the option to input specific files for analysis within the web tool's interface. 3. The web tool prompts the user to upload the desired file. 4. The user selects the file they want to upload to. 5. The web tool receives the uploaded file(s) and begins processing the data for the report generation. |
| Subflows:  The web tool processes the data contained within the uploaded file |
| Alternate Flows: |

|  |  |  |
| --- | --- | --- |
| Use Case Name: Report Generation for URL | ID: 3 | Importance: High |

|  |  |
| --- | --- |
| Primary Actor: User | Use Case Type: Essential |

|  |
| --- |
| Stakeholders:  **User**: Provides a clear dashboard that contains useful website information collected regarding the uploaded URL. |
| Brief Description: When a URL is uploaded and all the data is collected into an associated JSON, a template is dynamically filled with the relevant information. |
| Trigger: Valid URL is submitted by user Type: External / Temporal |
| Relationships:  Associated with User Input, URL Input, Sitemap Parser. |
| Normal Flow of Events:   1. Every function that returns data does so through a master JSON file. 2. A template using jinja2 is filled using the data values stored in the JSON file. 3. The template is presented to the user with all the information that was collected. |
| Subflows: |
| Alternate Flows:  If the JSON has empty values, or if an error has occurred, do not add the information to the dashboard template. |

|  |  |  |
| --- | --- | --- |
| Use Case Name: Tool Descriptions | ID: 4 | Importance: High |

|  |  |
| --- | --- |
| Primary Actor: User | Use Case Type: Essential |

|  |
| --- |
| Stakeholders:  **User**: Provide helpful information about the tools utilized to gather information that is available from the report. |
| Brief Description:  How the tool provides additional information to the web tool. |
| Trigger: Report Generation Type: External / Temporal |
| Relationships:  Associated with Report Generation |
| Normal Flow of Events:   1. When the data is collected by each tool during the report generation process, the web tool identifies the specific data collection tools used. 2. The web tool provides an option for the user to access detailed descriptions of each tool's functionalities. 3. The user selects the "Tool Descriptions" option from the user interface. 4. The web tool displays comprehensive information about the tool. |
| Subflows: |
| Alternate Flows: |

|  |  |  |
| --- | --- | --- |
| Use Case Name: Sitemap Parser | ID: 5 | Importance: High |

|  |  |
| --- | --- |
| Primary Actor: User | Use Case Type: Essential |

|  |
| --- |
| Stakeholders:  **User**: Receives all links associated with a website via the website’s sitemap.  This feature proves to be useful in various investigations, as finding important/hidden web pages can provide additional information. |
| Brief Description:  This use case describes the process of detection, extraction, and display of a website’s sitemap after the user has requested an URL to be analyzed. |
| Trigger: The user inputs a valid URL to the web analyzer Type: External / Temporal |
| Relationships:  Includes URL Input |
| Normal Flow of Events:   1. User submits a valid URL as input to the web analyzer module. 2. Check for the robots.txt file associated with the root page. 3. If robots.txt is found on the website, parse the sitemap URLs from the file. 4. Recursively traverse each sitemap associated with the website, while outputting each link associated with the website to a JSON file for future implementation. |
| Subflows: |
| Alternate Flows:  S-3: If robots.txt is not found on the website, return an error code. |