

Software Design Document for Social Butterfly

Version <1.1> Approved

Prepared by:

Group Name: Code Crafters

Juliana Unger
Matthew Henao
Tarek Kayali

z23618477
z23685608
z23710133

junger2021@fau.edu
henaom2022@fau.edu
Tkayali2023@fau.edu

Instructor: Safak Kayıkcı

Course: Principles Software Engineering

Section: 4010-002

Teaching assistant: Preston Billion Polak

Date: July 14, 2024

<p style="text-align: center;">Social Butterfly Revision Sheet</p>
--

Version	Revision Date	Changes	Author
1.1	7/13	Changed the intro and scope and table of contents	Juliana

Table of Contents

Social Butterfly Software Design Document.....	4
1.0 INTRODUCTION.....	4
1.1 Purpose.....	4
2.0 SYSTEM OVERVIEW.....	6
3.0 System Architecture.....	6
3.1 Architectural Design	6
3.3 Design Rationale.....	7
4.0 Data Design	8
4.1 Data Description.....	8
4.2 Data Dictionary	8
5.0 Component Design	9
5.1 Component Descriptions	9
6.0 Human Interface Design.....	11
6.1 Overview of User Interface	11
6.3 Screen Objects and Actions.....	12
7.0 Requirements Matrix.....	13
8.0 APPENDICES	14
Appendix A: Glossary of Terms	14
Appendix B: Diagrams and Tables.....	14
Appendix C: Code Snippets	15

Social Butterfly Software Design Document

1.0 INTRODUCTION

1.1 Purpose

The purpose of this Software Design Document (SDD) is to provide a comprehensive outline of the design and architecture for the Social Butterfly project. This Python-based tool is designed to empower users with the ability to collect, analyze, and derive insights from social media data, specifically from Twitter (X) and Reddit platforms. The document serves as a crucial reference point for developers, project managers, and stakeholders involved in the development and maintenance of the Social Butterfly system.

This SDD aims to:

1. Define the overall system architecture and component interactions.
2. Specify the design patterns and best practices to be employed.
3. Detail the data flow and processing mechanisms within the system.
4. Outline the user interface design and user experience considerations.
5. Provide a clear roadmap for implementation, testing, and future enhancements.
6. Ensure alignment between technical specifications and project requirements.
7. Facilitate communication and understanding among team members and stakeholders.

This document provides detailed information on the system's structure, components, interfaces, and other critical elements. It will guide the development process, ensure consistency in implementation, and serve as a foundation for future maintenance and upgrades.

1.2 Scope

The scope of the Social Butterfly project encompasses the development of a comprehensive, Python-based tool designed to revolutionize social media data analysis. This tool will provide users with an intuitive interface to search, collect, and analyze posts from Twitter

(X) and Reddit, leveraging the Tweepy and PRAW libraries for API interactions. The system's core functionalities include advanced search capabilities with customizable parameters, efficient data collection and storage mechanisms, sophisticated analysis tools featuring sentiment analysis and trend identification, and customizable report generation in various formats (e.g., PDF, CSV). Real-time data visualization will be implemented to represent trends and patterns graphically, enhancing user understanding of social media dynamics. The tool will incorporate user authentication and data privacy measures to ensure secure access and usage. While the current scope focuses on Twitter and Reddit integration, the architecture will be designed with scalability in mind, allowing for potential future expansion to additional social media platforms. The primary goal is to empower users across various sectors – from social media managers and market researchers to academic institutions and businesses – with actionable insights derived from online conversations and community discussions, ultimately facilitating data-driven decision-making in the rapidly evolving social media landscape.

1.3 Overview

This document is structured to provide a detailed understanding of the Social Butterfly project, including:

- System Overview
- Design Considerations
- Architectural Design
- Detailed Design
- Design Patterns and Best Practices
- Implementation Considerations
- Testing and Validation
- Maintenance and Support
- Appendices

1.4 Reference Material

- Twitter API Documentation: [Twitter API Docs](#)
- Reddit API Documentation: [Reddit API Docs](#)
- Tweepy Documentation: [Tweepy Docs](#)
- PRAW Documentation: [PRAW Docs](#)

1.5 Definitions and Acronyms

- API: Application Programming Interface
- Twitter (X): A social media platform for microblogging
- Reddit: A social media platform and forum for community discussions
- Tweepy: A Python library for accessing the Twitter API
- PRAW: Python Reddit API Wrapper, a Python library for accessing the Reddit API
- SDD: Software Design Document

2.0 SYSTEM OVERVIEW

Thus, the use of the Social Butterfly tool is an attempt to create a highly specialised software system for the modern world of social media data processing. Originally, at the heart of the system is the interaction with the API of the sources, namely Twitter and Reddit, using Tweepy and PRAW to extract real-time information according to the set queries and parameters. This data retrieval process should be designed for large volumes of information and, at the same, it should respect API rate limits in addition to checking and validating the information getting. Once data is fetched, the system uses the following multi-layered processing sequence in view of the fact that relevant information ingestion has to be meticulously accomplished by the system due to its sheer volume and constantly changing and updating nature. The raw data is processed to make basic standardizations before the data is imported to the various platforms. Such standardized data is then stored in a very efficient and reliable database system that facilitates the retrieval and analysis of the information. Through specific algorithms, the tool employed in analysis considers natural language processing for sentiment analysis and statistical modelling techniques for trend analysis, as well as calculates engagement metrics.

The reporting module acts as a tool that turns this analytical information into something easy to understand and present in the form of very attractive graphical representations. For their part, users are provided with the ability to filter data for report generation to suit their needs with options like heat map reports, trend graph reports and reports showing the distribution of positive/negative sentiments, among others. The system also has a revolving dashboard capacity to check trends in social media as they enhance. This is due to the intrinsic security features of the system, where issues in user authentication, data protection, and the generally applicable data protection laws have been taken care of. Its modularity also allows the tool to grow with the organization and easily accommodate future expansions like adding more social networks and utilizing more sophisticated machine learning algorithms for predictive purposes.

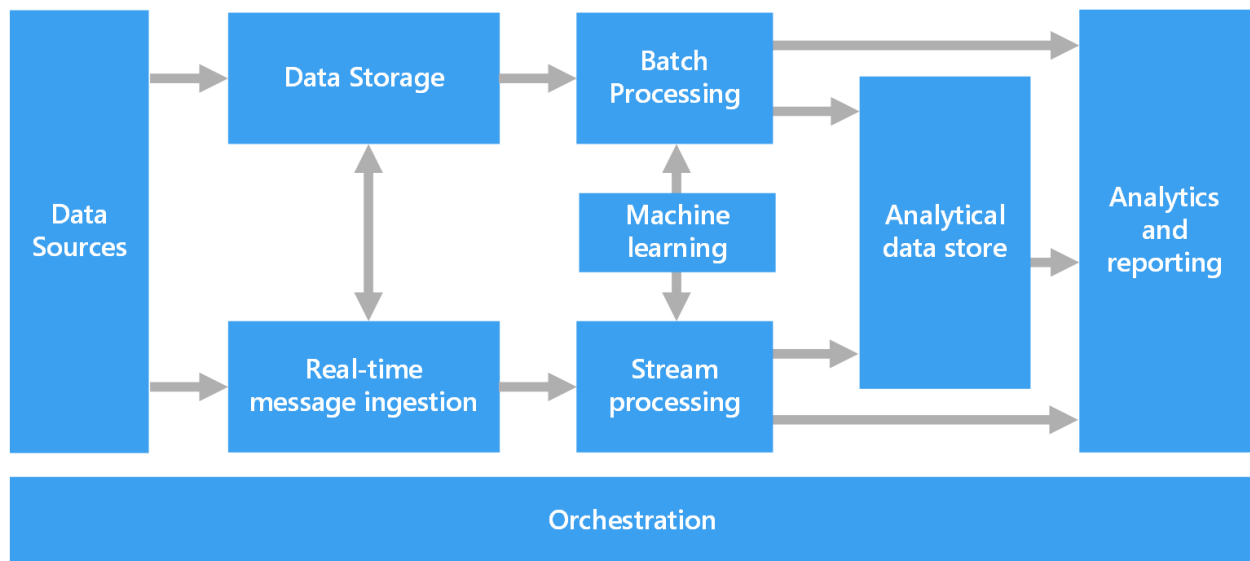
3.0 System Architecture

3.1 Architectural Design

The architecture of Social Butterfly includes several key subsystems:

- User Interface: Allows users to input search parameters and view results.
- Data Retrieval: Fetches data from Twitter and Reddit APIs.
- Data Storage: Stores the retrieved data in a structured format.
- Data Analysis: Analyzes the collected data to provide insights.
- Reporting: Generates reports based on the analysis.

3.1.1 Subsystem Interactions



3.2 Decomposition Description

- User Interface: Provides search and reporting functionalities.
- Data Retrieval: Uses Tweepy and PRAW to fetch data.
- Data Storage: Manages data storage using a database.
- Data Analysis: Processes and analyzes the data.
- Reporting: Generates and formats reports.

3.3 Design Rationale

The design is structured to ensure modularity, scalability, and maintainability. Each subsystem is designed to handle specific tasks for easy development and future updates.

4.0 Data Design

4.1 Data Description

Since Social Butterfly is intended to be a data-driven application, it is necessary to describe the data design for the application, which will mean storing user information, social media posts, and analysis results in a structured manner within a database. This database is intended for the optimization of the key types of data received and analyzed by the system. It entails account details and records, preferences, and search histories, which enhances users' experiences. Data from Twitter (X) and Reddit in the form of raw social media posts are saved together with the complete text of the post, post-meta-information, and its corresponding metrics, such as the number of likes, retweets, or upvotes. This part is linked to their original location and authors for easier reproach and data analysis. It should also be noted that outcome examples of analysis are very versatile and include such aspects as positive and negative tones, observed tendencies, users' activity rates, and time shifts. The data model of the database is particularly designed to support fast queries and searches in order to produce data reports on the fly. To support scalability, data design also involves proper indexing and partitioning of the data, which in turn enables the system to manage large amounts of data involved in social media. This database structure also has the advantage of data integration, interconnection of different data fields and highly effective options for analyzing the information.

4.2 Data Dictionary

- Users: Stores user information and preferences.
- Posts: Stores retrieved posts from Twitter and Reddit.
- Analysis: Stores results of data analysis.

5.0 Component Design

5.1 Component Descriptions

5.1.1 Data Retrieval Component

Description:

The Data Retrieval Component uses the Tweepy and PRAW libraries to collect data from Twitter (X) and Reddit. It links Social Butterfly with social media APIs to implement valid and efficient data extraction processes.

Key Responsibilities:

- Authentication with APIs
- Formation of a query from the data entry of a customer
- Rate limit management
- Error handling
- Data parsing

Methods:

- `fetch_twitter_data()`:

Purpose: This pulls information from Twitter using the Tweepy.

Parameters: query, count, lang, result_type, since_id, max_id

Process:

- a. Credentialed to use the Twitter API
- b. Constructs search query
- c. Carries out an API call, manages paginating
- d. Filters tweets and pulls data from them
- e. Manages rate limiting

Returns: List of structured tweet data of firms that have reported their Q2 2018 results before the market close on 24 th July 2018.

Return API errors and network problems

- `fetch_reddit_data()`:

Purpose: Fetches data from Reddit with the help of PRAW.

Parameters: `subreddit`, `sort`, `time_filter`, and `limit`

Process:

- a. Creates Reddit API manager
- b. Enters the specified subreddit with the sorting/filtering option
- c. Gets post data and, conditionally, comments

Another practice that adheres to the API usage rules is(Figure 3):

Returns: The structured post that I found and the corresponding comment

Admittedly handles all regards subreddit access problems and API glitches

Additional Features:

- The use of a cache mechanism, especially for the most frequently used queries
- Logging functionality
- Versatility for any future addition of features to the platform
- API compatibility updates for the library

This component guarantees the correct data acquisition from the Twitter and Reddit platforms, which are the basis of Social Butterfly's analytical and reporting capabilities.

5.1.2 Data Storage Component

Finally, the Data Storage Component is an inevitable component of the Social Butterfly system as it is in charge of organizing the retrieval and storing of data in a structural format of social media. They guarantee the proper storage, fast availability and stable durability of the information that is gathered from the Twitter microblogging site and the Reddit social news aggregator. It primarily operates through two key methods: functions to store and retrieve the data. The `store_data()` function is intended to accept the data from the Data Retrieval Component, prepare it for storing, merge it when conflicting and necessary, and insert a merged record into the database if it isn't a duplicate. The other method is the `retrieve_data()`, which enables the pulled-out information from the database depending on the query or parameters placed in the system for analysis and other reports. Through these methods, the Data Storage Component works as the coordinating point where tracking, and sometimes analyzing as well as

presenting the Social Butterfly info can be done in support of the application's data collection, analysis, and presentation phases.

5.1.3 Data Analysis Component

The Data Analysis Component is the control centre of Social Butterfly as it analyzes social media information that is gathered. It employs two primary methods: The two examined functions include the `analyze_sentiment()` and `generate_trends()`. The `analyze_sentiment()` method uses text analysis on posts to determine their sentiment as either positive, negative, or neutral. The method `*generate_trends()*` pinpoints the frequently discussed subjects, hashtags, and tendencies in the analyzed data, helping to get insights into the communication processes and users' actions. This component converts crude information into valuable insights so that the clients can assess the trend in people's perception towards any topic across multiple social media channels.

5.1.4 Reporting Component

The Reporting Component is the last step within the University's Social Butterfly process, where gathered and analyzed data are converted into meaningful and presentable formats. It offers two main methods: The second and third function definitions are `generate_pdf_report()` and `generate_csv_report()`. The next useful method is `generate_pdf_report()`, which generates HTML formatted PDF containing the data analysis in the form of figures, tables, and statistical benchmarks. The `generate_csv_report()` function generate csv formatted report, to provide organized format to data that can be exported in other tools. This component makes sure that all the analytic work that is conducted on the social media data is presented to the user in the form of easy to understand formats, tools, and dashboards that can be used in the formulation of social media strategies and decisions.

6.0 Human Interface Design

6.1 Overview of User Interface

In the Human Interface Design of Social Butterfly, focus is directed toward ease of use and operability and providing both versions of the application for usage regardless of the user's experience level. The design of the interface is characterized by rather minimalist and sleek look, which is further accompanied by an optimized and flexible layout. The primary interface of the tool includes the feed that contains information about the latest analyses, popular topics, and shortcuts to frequently used actions. An extensively visible search bar is provided where users need to put the keywords, hashtags or the particular social media profiles to be investigated. Further search options can be set by clicking on the '+' symbol on the top right corner of the panel which offers fine-tuning of the options for data display. Last but not the least, data visualization is one of the components here and charts or graphs that are incorporated are live that means they update themselves in terms of analyses results. The report generation interface is equipped with templates and form items that can be easily moved around the prepared forms. In this case, a permanent bar is located at the bottom of the work area and contains links to all sections of the application: saved searches, personal settings, and manuals. This means that the color pallet is chosen in order to minimize eye fatigue when one has to spend a lot of time with the app, which comes in both light and dark versions. _Tooltips and contextual help bubbles_

keep users from getting lost in minor sub-features and help them learn the product's functionalities easily. In summary, the design of Social Butterfly concentrates on enhancing usability because the application's purpose is to assist the user in managing, as well as interpreting and presenting the data, originating from social networks, as quickly and easily as possible.

6.2 Screen Images



Positive Words - Wordcloud

6.3 Screen Objects and Actions

A clear and integrated user interface of the Social Butterfly application embraces a comprehensive layout of the application's interface as it seeks to operate under a simplistic mode that would enable customers to interact with the application effortlessly in the multiple aspects that the application holds. The Search Interface is the first point of contact. It incorporates textual boxes where a user sets the keywords, drop-down list where a user sets options like date range and social media site, and a large prominently located search button. It is very easy for them to enter their parameters of search and they are able to launch a search operation with a click of a button. Automatically, users switch to the Report Generation Interface where they find the interface has been conveniently laid to suit the output of the analysis of their choice. This section consists of format choice, it is the choice of types of report users want and, in addition to that, a full range of the set of versatility of analyses, depth, and focus of the reports. Okay, on the top of the screen, a generate button will wait to sum up all the data chosen and put it in the format selected. This is a crucial part of the application as it is a control panel where summarized data and various charts and graphs are placed to give a brief vision of the users' result of social networking. A refresh button helps the user to refresh the concomitant view at anytime to reflect on the new data. In the entire process of these interfaces, labelled and navigable screen objects

enable the effective and easy interaction and clearly defined actions help users completing the process of searching and reporting of the social media trends and sentiments.

7.0 Requirements Matrix

The requirements matrix provides a mapping between the functional requirements and the corresponding design elements.

Requirement	Design Element
Data Retrieval	Data Retrieval Component
Data Storage	Data Storage Component
Data Analysis	Data Analysis Component
Report Generation	Reporting Component
UI (User Interface)	User Interface
Error Handling	Error Handling Mechanisms

8.0 APPENDICES

Appendix A: Glossary of Terms

This section should contain a list of technical terms and their definitions.

- API (Application Programming Interface): A set of protocols and tools for building software applications.
- Sentiment Analysis: The process of computationally identifying and categorizing opinions expressed in a piece of text.
- Data Visualization: The graphical representation of information and data.
- Tweepy: A Python library for accessing the Twitter API.
- PRAW: Python Reddit API Wrapper, a Python package that allows for simple access to Reddit's API.

Appendix B: Diagrams and Tables

Table B1: Social Butterfly Database Schema

Table Name	Column Name	Data Type	Constraints	Description
Users	user_id	INTEGER	PRIMARY KEY	Unique identifier for each user
	username	VARCHAR(50)	NOT NULL, UNIQUE	User's chosen username
	email	VARCHAR(100)	NOT NULL, UNIQUE	User's email address
	password_hash	VARCHAR(255)	NOT NULL	Hashed password for security
	created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	User account creation date
Posts	post_id	INTEGER	PRIMARY KEY	Unique identifier for each post
	platform	VARCHAR(10)	NOT NULL	Social media platform (Twitter/Reddit)
	content	TEXT	NOT NULL	The text content of the post

	author	VARCHAR(50)		Author of the post
	posted_at	TIMESTAMP		Time when the post was created
	engagement_count	INTEGER		Number of likes/retweets/upvotes
Analysis	analysis_id	INTEGER	PRIMARY KEY	Unique identifier for each analysis
	post_id	INTEGER	FOREIGN KEY	Reference to the analyzed post
	sentiment	VARCHAR(10)		Sentiment of the post (positive/negative/neutral)
	score	FLOAT		Sentiment score
	analyzed_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Time of analysis
Searches	search_id	INTEGER	PRIMARY KEY	Unique identifier for each search
	user_id	INTEGER	FOREIGN KEY	Reference to the user who performed the search
	query	VARCHAR(255)	NOT NULL	The search query
	platform	VARCHAR(10)	NOT NULL	Platform searched (Twitter/Reddit/Both)
	search_date	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Time o

Appendix C: Code Snippets

1. Twitter Data Retrieval:

```
import tweepy

def fetch_twitter_data(query, count=100):
    # Twitter API credentials here
    auth = tweepy.OAuthHandler("consumer_key", "consumer_secret")
    auth.set_access_token("access_token", "access_token_secret")

    api = tweepy.API(auth)

    tweets = api.search_tweets(q=query, count=count)
    return [tweet._json for tweet in tweets]
```

2. Reddit Data Retrieval:

```
import praw

def fetch_reddit_data(subreddit, limit=100):
    reddit = praw.Reddit(client_id="your_client_id",
                        client_secret="your_client_secret",
                        user_agent="your_user_agent")

    subreddit = reddit.subreddit(subreddit)
    posts = subreddit.hot(limit=limit)
    return [post.__dict__ for post in posts]
```

3. Sentiment Analysis:

```
from textblob import TextBlob

def analyze_sentiment(text):
    analysis = TextBlob(text)
    if analysis.sentiment.polarity > 0:
        return 'positive'
    elif analysis.sentiment.polarity == 0:
        return 'neutral'
    else:
        return 'negative'
```