

Planned Sprint	Actual Sprint	US ID	User Story Description
S1	S1	US-001	Environment Setup & Repository Initialization
S1	S1	US-002	Setup Development Environment & Dependencies
S2	S2	US-003	Integrate Google News API Data Fetching
S2	S2	US-004	Integrate Twitter API for Recent Tweets
S3	S3	US-005	Implement VADER Sentiment Analysis for News Data
S3	S3	US-006	Implement VADER Sentiment Analysis for Twitter Data
S4	S4	US-007	Normalize, Concatenate, and Clean Data
S4	S4	US-008	Aggregate Sentiment Data into Time Buckets
S5	S5	US-009	Develop Sentiment Forecasting using Prophet
S6	S6	US-010	Build Interactive Streamlit Dashboard Layout
S6	S6	US-011	Implement Real-time Alerts for Sentiment & Volume Spikes
S7	S7	US-012	Add CSV Download Buttons for Raw, Combined, Aggregated, Forecast Data
S7	S7	US-013	Add Cache & Data Refresh Mechanism
S8	S8	US-014	Improve Error Handling & Logging for API Requests
S8	S8	US-015	Add Keyword Trend Analysis & Visualization
S9	S9	US-016	Write Comprehensive Documentation & Developer Guide
S9	S9	US-017	Conduct Unit & System Integration Tests
S10	S10	US-018	Set up CI/CD for Deployment to Streamlit Sharing or Cloud

S10	S10	US-019	Perform Final User Acceptance Testing and Feedback Incorporation
-----	-----	--------	------------------------------------------------------------------

MoSCoW	Dependency	Assignee	Status
MUST HAVE	VS Code, Git	Shlok Gautam	Completed
MUST HAVE	Python 3.8+, pip	Shlok Gautam	Completed
MUST HAVE	GNews Client, API keys	Shlok Gautam	Completed
MUST HAVE	Tweepy, Twitter API keys	Shlok Gautam	Completed
MUST HAVE	NLTK, pandas	Shlok Gautam	Completed
MUST HAVE	NLTK, pandas	Shlok Gautam	Completed
MUST HAVE	pandas, numpy	Shlok Gautam	Completed
MUST HAVE	pandas	Shlok Gautam	Completed
SHOULD HAVE	Prophet	Shlok Gautam	Completed
MUST HAVE	Streamlit, plotly	Shlok Gautam	Completed
MUST HAVE	Slack API, requests	Shlok Gautam	Completed
COULD HAVE	Streamlit	Shlok Gautam	Completed
SHOULD HAVE	Streamlit cache	Shlok Gautam	Completed
SHOULD HAVE	Python logging	Shlok Gautam	Completed
SHOULD HAVE	pandas, plotly	Shlok Gautam	Completed
MUST HAVE	Markdown, ReadTheDocs	Shlok Gautam	Completed
MUST HAVE	pytest, streamlit	Shlok Gautam	Completed
MUST HAVE	GitHub Actions, dotenv	Shlok Gautam	Completed

MUST HAVE	User feedback	Shlok Gautam	Planned
-----------	---------------	--------------	---------

Acceptance Criteria	Tasks/Comments
Repo initialized with README, .gitignore; basic folder structure setup.	Setup repo, add initial docs, configure git hooks.
Virtual environment active; dependencies in requirements.txt installed.	Create venv, requirements, install packages.
Pull news articles by search query; data frame structured correctly.	API key management, fetch loop, error handling.
Fetch recent tweets matching keywords and store in DF.	Pagination, user mapping, rate-limit handling.
Sentiment scores and labels added to news data frame.	Install nltk, mocking for tests.
Sentiment analysis integrated for tweets similarly.	Create reusable sentiment functions.
Unified data with consistent schema; missing data handled.	Timestamp parsing & timezone normalization.
Aggregated metrics (mean sentiment, count) per hour/day with smoothing window applied.	Implement rolling averages, z-score computations.
Produce sentiment forecasts with confidence bounds.	Model fitting, future frame prediction, visualization prep.
Dashboard with sidebar controls, tabbed views, responsive charts.	Design UX/UI, implement navigation, data binding.
Alerts triggered and sent via Slack webhook integration.	Define alert rules, threshold sliders, test webhook message format.
User able to export datasets as CSV files from each dashboard view.	Implement Streamlit download feature, optimize CSV conversion.
Data refresh on-demand with caching to reduce API calls and speed interface.	Use decorators, manage cache invalidation.
Graceful handling of API errors, retries, and logging for diagnostics.	Add logging, retry logic, fallback defaults.
Keywords extracted, counted, visualized with spikes and z-scores.	Keyword regex matching, plotly line charts.
Full README and guides for setup, use, and contribution available.	Document code, API usage, environment setup and dashboard walkthrough.
Tests created, run automatically in CI environment.	Write unit tests for key modules; setup test suite.
Successful build and deployment workflow with environment variables protected.	Configure workflows, secrets, deployment scripts.

User feedback collected and addressed prior to final release.	Schedule demo sessions, collect bugs/improvements, document fixes.
---------------------------------------------------------------	--------------------------------------------------------------------