

Case study - Vortex Media

Project Name:	Vortex Media
Project Duration:	2 years, ongoing project
URL:	https://vx.yam.md , https://news.yam.md/ro/subjects
Current Status:	Launched in 2017

Vortex Media is a software platform which monitors and analyses online news coverage. It uses sophisticated background technology to extract and process content from 99% of important media outlets available online in Moldova.

The challenge

The client approached Atomate to help develop a high availability media monitoring solution focused on data aggregation, machine learning classification, clustering and further analysis with a complex user accessible frontend. Please note that the process is ongoing as the development continues.

The process

We engaged with the client to perform the following activities

- Discovery
 - Organising a series of workshops with the client to gather initial requirements
 - Develop implementation master plan
 - Develop specifications for each phase, including UX flows and wireframes
 - Develop solution development and testing guidelines, as well as solution architecture
- Solution implementation, phased approach
 - Around 4 phases per year
 - Team composition: 2-3 developers (Backend / Frontend), 2 QA developers, 1 PM
 - Test automation
- Solution stabilisation & refactoring
 - Solution stabilisation by continuous testing using manual and automated approach
 - Support for current production release, while the next one is in development.

Notes on technology stack and solution architecture

The following technology stack has been used to implement the project.

Component	Framework / Technology
Frontend	<ul style="list-style-type: none"> ● React JS / Backbone Javascript Library for UI / UX ● PHP / Laravel web framework for additional functionality ● Frontend handling up to 50 requests per seconds ● Search mechanisms based on Apache SOLR faceting, quick and efficient ● Billing integration with Stripe and Paypal ● Mailgun API integration for emailing

Vortex Data Admin Application	<ul style="list-style-type: none"> • Spring Boot application with web sockets integration for real time job tracking; REST API integration with Vortex Data aggregation module • Multi level user access with various permissions with various Admin functions. • Stripe / Paypal integration
Vortex Data aggregation module	<ul style="list-style-type: none"> • Java 8 / Spring MVC with scheduling component with advanced job handling engine for efficient job management and execution. • Data aggregation module, with decision algorithms around broken data extraction feeds and pipes. Ability for the system to self-heal. • Robust implementation of complex resource hungry methods using AWS Lamda functions • REST API with security layer for data extraction
Database & Storage	<ul style="list-style-type: none"> • Mysql Master / Slave setup, fine tuned for performance, with replication • Redis for key/value storage • AWS S3 storage for files • Apache SOLR for Search API, also used in data hungry algorithms
Server architecture	<ul style="list-style-type: none"> • Load balanced, multi-server setup • Scalable and flexible server architecture, currently handling 50 requests per seconds and 5GB/day incoming data processing, including social media data extraction • Automated scaling configured for spikes in activity
Development environment	<ul style="list-style-type: none"> • Gitflow Workflow, code review via code pull requests • Atlassian JIRA / Confluence • Continuous integration using Jenkins • Test automation using Robot Framework / Selenium • Docker based development approach • Multiple server development environments, resembling as much as possible

Notes on algorithms

Classification	<p>The team used classification algorithms to create an automated classifier for incoming data, based on agreed predefined categories. Each category has training data sets available.</p> <ul style="list-style-type: none"> • Naive Bayes, Multinomial Naive Bayes • Support Vector Machine with Ngrams • In combination with Eigenfaces algorithm used in face problems detector.
Clusterization	<p>The team used clusterization algorithms to detect clusters of similar data.</p> <ul style="list-style-type: none"> • K-Means - a method of vector quantization, popular for cluster analysis in data mining.

Sentiment Analysis	Used in stories sentiment detector: <ul style="list-style-type: none">• The team implemented sentiment analysis using Naive Bayes, but also had versions running with Support Vector Machine
Named Entities	Used to detect names of places, people and organisations within news stories. Connected with Sentiment Analysis to determine sentiment for each personality in a story. <ul style="list-style-type: none">• Support Vector Machine with Ngrams

Business value

Following the successful launch of Vortex Data platform, users praised the system's features and intuitive design. It's been reported that 99 percent of Vortex data customers who have purchased accounts work with it every day. Together with Atomate, Vortex Data is continuing development of their software.