

Yuri Zhylyuk

<https://linkedin.com/in/zhylyuk>

Selected data visualisations: <http://yznotes.com/portfolio>



Selected Data Analytics Projects

BI Solution for FENTY brand at LVMH Group, France

LVMH Group, France, Mar 2019 - June 2020

End-to-end development and implementation of a custom BI solution for new multi-channel brand (Ecommerce, Retail, Wholesale), including technical design, architecture, infrastructure, ETL, data visualisation and reporting.

CRM Analytics solution

L'Oréal Australia, Feb 2018 - March 2019

Integration of data from various internal and external data sources (Salesforce Marketing Cloud, CheckMarket, Google Analytics, Google Drive, Retail Sales, etc.) into DOMO BI platform, ETL/ELT using DOMO Redshift SQL and Python API. Automated monthly KPI calculation and historical tracking for over 20 individual KPI indicators across multiple brands. Design and development of dashboards.

Net Promoter Score data integration and reporting

L'Oréal Australia, May 2018 - Dec 2018

Data integration of post purchase survey data for selected L'Oréal Luxury brands into DOMO BI platform, automated reporting and monitoring, custom portal for staff for tracking and follow up.

BI and Analytics Platform 'Glinda'

MECCA Brands, Mar 2017 – Jul 2017

Architecture, design and technology lead for BI and Analytics Platform 'Glinda', including Data Integration Engine, Data Storage and Processing Engine (Based on AWS + Apache Airflow + PostgreSQL + Python/Dask).

Mecca Beauty Loop (Customer Loyalty) Engine

MECCA Brands, Nov 2016 – Feb 2017

Infrastructure, calculation engine and internal CRM application for MECCA Brands loyalty program.

● Python ● Anaconda Dask ● PostgreSQL ● AWS RDS

MECCA Datalab

MECCA Brands, Jan 2016 – Mar 2016

Championed, designed and implemented 'MECCA Datalab' environment (based on Jupyterhub and Bokeh Server) for reproducible exploratory data analysis, data mining, visualisation and reporting.

Real Time Sales Analytics

MECCA Brands, Nov 2015 – Dec 2015

Custom analytics application that asynchronously collects, aggregates and visualises data from about 80 stores (individual Oracle databases) across Australia and New Zealand. All calculations and charts are refreshed with every new transaction.

● Python ● RethinkDB ● REST API ● Web Sockets ● JavaScript ● Raspberry Pi (as a client - attached to a TV screen)

System Jobs and Interfaces Monitor

University of Melbourne, Apr 2015

Application that collects information about running system jobs, builds a multidimensional dashboard with cross-filtering functionality for exploratory analysis and sends email alerts about failed/problematic jobs and data feeds. Application runs 24/7 with 10 minutes intervals, Amazon Web Services used for web hosting and emailing alerts. This application saves time of data analysts on a daily basis and prevents system failures in a proactive manner.

Data Entry Robot

University of Melbourne, Mar 2015

Set of scripts (based on Python pywinauto and image recognition libraries) for automated unsupervised data entry. This solution was used for making bulk ad hoc changes in the systems (via GUI) and in many cases eliminates a need of employing and training casual data entry staff, which saves significant amount of time and money.

Automated Data Entry Auditing

University of Melbourne, Dec 2014

Automated audit scripts and reporting for monitoring system configuration changes made by data entry staff during Student System Reorganisation project. This solution allowed identifying data entry errors regularly and at early stages and significantly improved outcome of the project.

HTML HELP Builder

University of Melbourne, Nov 2014

Designed and developed a solution for converting System Help Documentation in MS Word files into HTML help modules which eliminated manual processing. Each Word document, that represents help module, is converted to hundreds of HTML files (based on Word document TOC) with all external URLs and internal links between modules being created and/or validated and XML menu files rendered automatically.

DataViewer

University of Melbourne, Jan 2014

Internal web application for navigating through student application data received from VTAC (more than 700 data fields for each record) and identifying potential duplicates (using 'fuzzy' matches with existing records in Student System Database). This application saves about 2 hours of a junior data analyst time daily and provides higher accuracy in identifying potential duplicates.