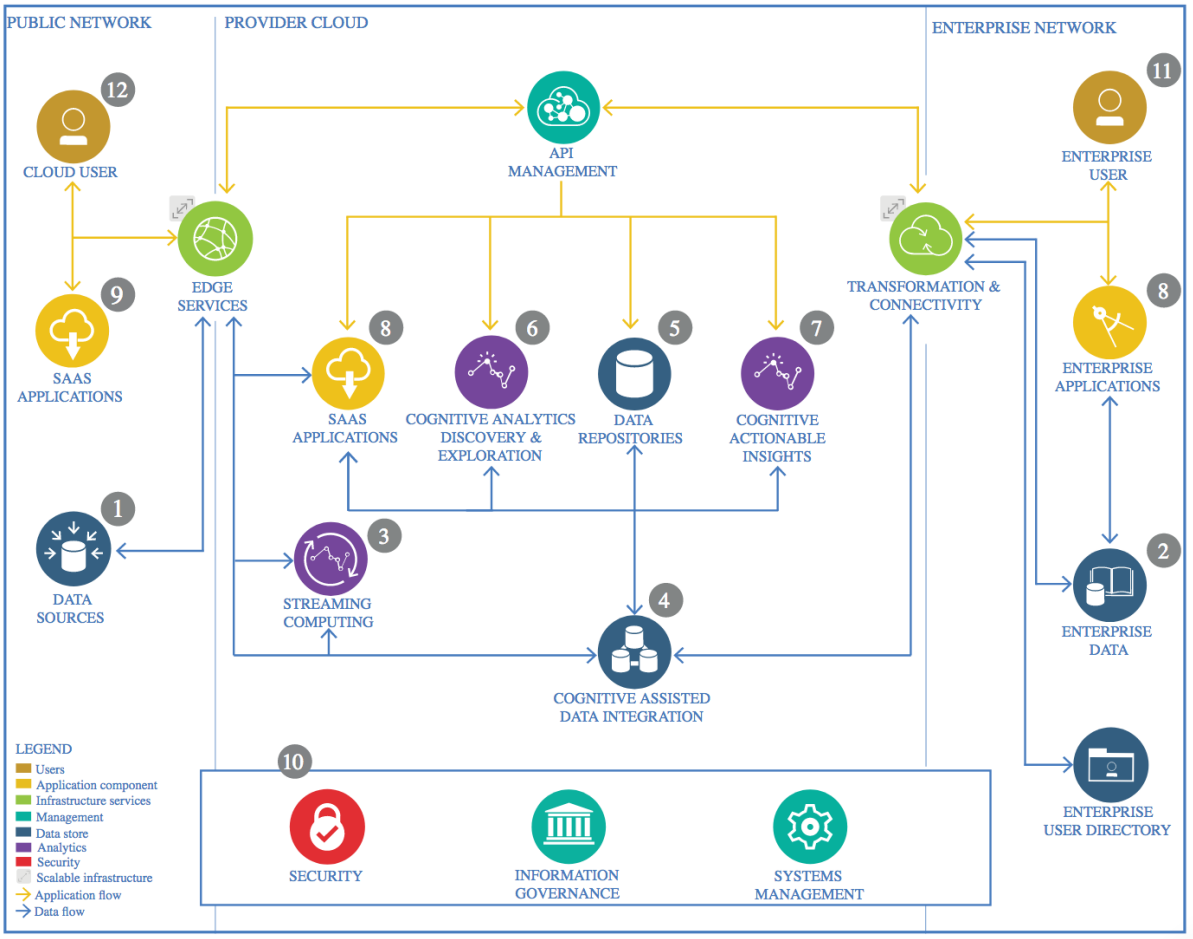
The Lightweight IBM Cloud Garage Method for Data Science

Architectural Decisions Document Template

1 Architectural Components Overview



IBM Data and Analytics Reference Architecture. Source: IBM Corporation

1.1 Data Source

1.1.1 Technology Choice

Pandas to download the data into the system.

1.1.2 Justification

CSV file with the timeseries and the data needed.

1.2 Enterprise Data

1.2.1 Technology Choice
Please describe what technology you have defined here. Please justify below, why. In case this component is not needed justify below.

1.2.2 Justification
Not needed

1.3 Streaming analytics

1.3.1 Technology Choice
Please describe what technology you have defined here. Please justify below, why. In case this component is not needed justify below.

1.3.2 Justification
Not needed

1.4 Data Integration

1.4.1 Technology Choice
Pandas,numpy,sklearn.

1.4.2 Justification
Pandas and Numpy were selected as it had a simple interphase and a big community where you can find help if you needed any. It is flexible and easy to use on data analysis and manipulation. Also, the dataset is not big so we don't need parallelization.
Sklearn provides easy tools to impute features and do further analytics.

1.5 Data Repository

1.5.1 Technology Choice
Ninja.Renewables

1.5.2 Justification
It is easy to intergrate and call from the api or download the data to csv file .It ensures you can save and load any relatable information you need on the case study.

1.6 Discovery and Exploration

1.6.1 Technology Choice
Pandas,Matplotlib,Seaborn

1.6.2 Justification

Matplotlib and Seaborn are common and handy tools for data visualization. They got a lot of build-in functions to plot correlation matrixes, histograms, scatter plots etc.

1.7 Actionable Insights

1.7.1 Technology Choice

Jupyter Notebook

1.7.2 Justification

You can run each cell separately , also easy to use interphase.

1.8 Applications / Data Products

1.8.1 Technology Choice

Please describe what technology you have defined here. Please justify below, why. In case this component is not needed justify below.

1.8.2 Justification

The conclusion of the research that was made is that the models can be used in a Solar Park so you can predict a 48hour ahead production of the Photovoltaic panels and then through a smart grid decide the amount of energy you will save, include into the system or dump.

1.9 Security, Information Governance and Systems Management

1.9.1 Technology Choice

Please describe what technology you have defined here. Please justify below, why. In case this component is not needed justify below.

1.9.2 Justification

Security should be maintained on the analytics side to avoid information leakage on the Solar Park information (name of the owner, location , total production and etc.

2 Development Process

2.1 Why have I chosen a specific method for data quality assessment?

The chosen method for data quality assessment primarily involves visual inspections and statistical analysis using plots and descriptive statistics. The following steps were taken:

- **Visual Inspection**: This includes plotting the data to identify any obvious anomalies or patterns.
- **Statistical Analysis**: Descriptive statistics (mean, median, standard deviation) are calculated to understand the data distribution and check for any outliers or inconsistencies.

- **Missing Data Handling**: Handling missing data by imputing or removing them to ensure the dataset's integrity and prevent issues during model training.

These methods were chosen because they provide a clear and comprehensive understanding of the dataset's quality, allowing for appropriate preprocessing steps to be applied.

2.2 Why have I chosen a specific method for feature engineering?

Feature engineering involved the following methods:

- **Temporal Features**: Sine and cosine transformations for 'Day' and 'Hour' to capture cyclical patterns in time-series data.
- **Normalization**: MinMaxScaler was used to normalize both features and target variables to ensure they are on the same scale, which is crucial for neural networks.

These methods were chosen because:

- Temporal transformations (sine and cosine) effectively capture the cyclical nature of the data.
- Normalization helps in speeding up the convergence of the neural network training process and prevents issues related to varying scales of input features.

2.3 Why have I chosen a specific algorithm?

The project employs three Artificial Neural Networks (ANNs) and one Long Short-Term Memory (LSTM) network. The choice of these algorithms is driven by their strengths in handling time-series data and capturing complex patterns:

- **ANNs**: Suitable for capturing non-linear relationships between input features and the target variable. ANNs are straightforward and effective for general prediction tasks.
- **LSTM**: Specifically designed for sequence prediction problems and excels at capturing long-term dependencies in time-series data.

2.4 Why have I chosen a specific framework?

The chosen framework is TensorFlow with Keras API:

- **TensorFlow**: A powerful and flexible deep learning framework that provides extensive support for model training, evaluation, and deployment.
- **Keras API**: Simplifies the construction and training of neural networks, making it user-friendly and efficient for rapid experimentation.

These frameworks were chosen for their robust community support, extensive documentation, and flexibility in model building and optimization.

2.5 Why have I chosen a specific model performance indicator?

The performance indicators used are:

- **Root Mean Squared Error (RMSE)**: Measures the average magnitude of errors. It is sensitive to large errors, making it useful for detecting significant deviations.
- **Mean Absolute Error (MAE)**: Measures the average magnitude of absolute errors. It is less sensitive to outliers compared to RMSE.
- **R-squared (R^2)**: Indicates the proportion of variance in the dependent variable that is predictable from the independent variables.

These indicators were chosen because they provide a comprehensive evaluation of the model's predictive performance, capturing different aspects of error and goodness-of-fit.

2.6 Data spliting

Data Splitting: The data is split into training and testing sets without shuffling to preserve the temporal order. This is crucial for time-series forecasting to ensure that future data points are predicted based on past data, simulating real-world scenarios.

2.7 Optimizer

Optimizer Choice: The use of RMSprop optimizer with a specific learning rate was chosen for its effectiveness in training neural networks, especially in dealing with the non-stationarity of the data.

2.8 Visuals

Visualization: Extensive use of plotting actual vs predicted values and performance metrics for each model. Visualizations help in intuitively understanding model performance and identifying areas for improvement.