

The background of the slide is a detailed historical map. The left side shows the continent of Africa with labels for the Abyssinian Highlands, Lake Rudolf, Lake Tanganyika, Lake Nyasa, Lake Bangweulu, Victoria Falls, and the Kalahari Desert. The right side shows Southeast Asia, including the Malay Peninsula, Sumatra, Java, and the Indonesian archipelago with islands like Celebes and Borneo. The map includes geographical features like the Gulf of Aden, the Indian Ocean, and the Strait of Siam. A white rectangular box with a thin blue border is centered over the map, containing the project title and member names.

Term Project

Geographic App for School Kids

Group Members:

Catalina Ramos, Geyang Zhang,

Nikolaos Tavladorakis, Ping-Hsuan Lin



Background and definition of the business use case

Our goal is to develop an app with geography information of all countries that can be used in schools to teach kids the most important geographical facts about each country. For example, the app will show the Top 10 geographical features (i.e., mountains, rivers, lakes, canyons, forests) of the selected geographical region. The app will function as follows:



The users select one or more geographical regions that they are interested in (i.e., Asia, USA, combination of countries, etc).



The app filters the database for the selected regions and returns the Top 10 of each geographical feature.



The app provides basic information and photos for the returned geographical features.

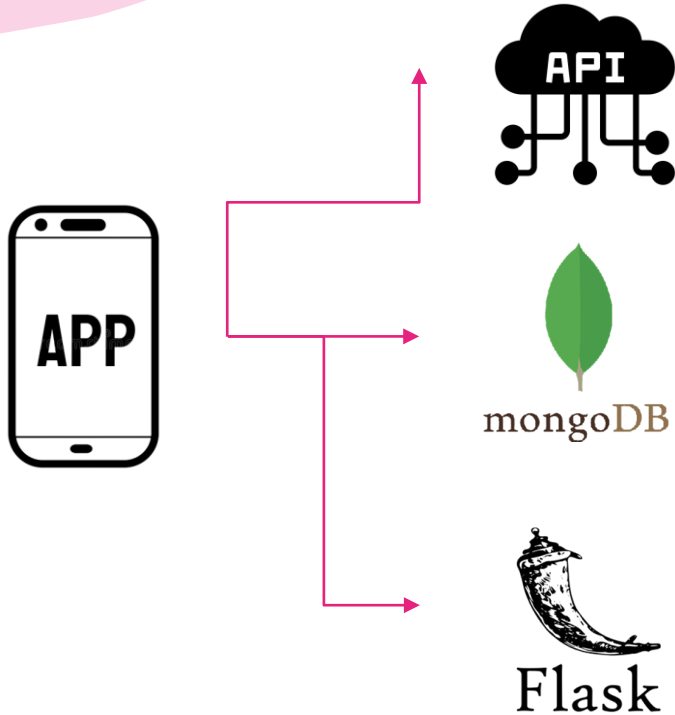
Data source specification and procurement details

The data for our application would consist of all the geographical features and relevant information from all the countries. The data source for all the required data would be the Wikipedia site. The procurement process is the following:

1. The Wikipedia python library connects with the Wikipedia website through their API, <https://en.wikipedia.org/w/api.php>.
2. It returns the URL for the selected country and geographical feature.
3. It scrapes the selected URL and returns a JSON file.
4. The JSON file is saved in the MongoDB.



Proposed design choices and rationale

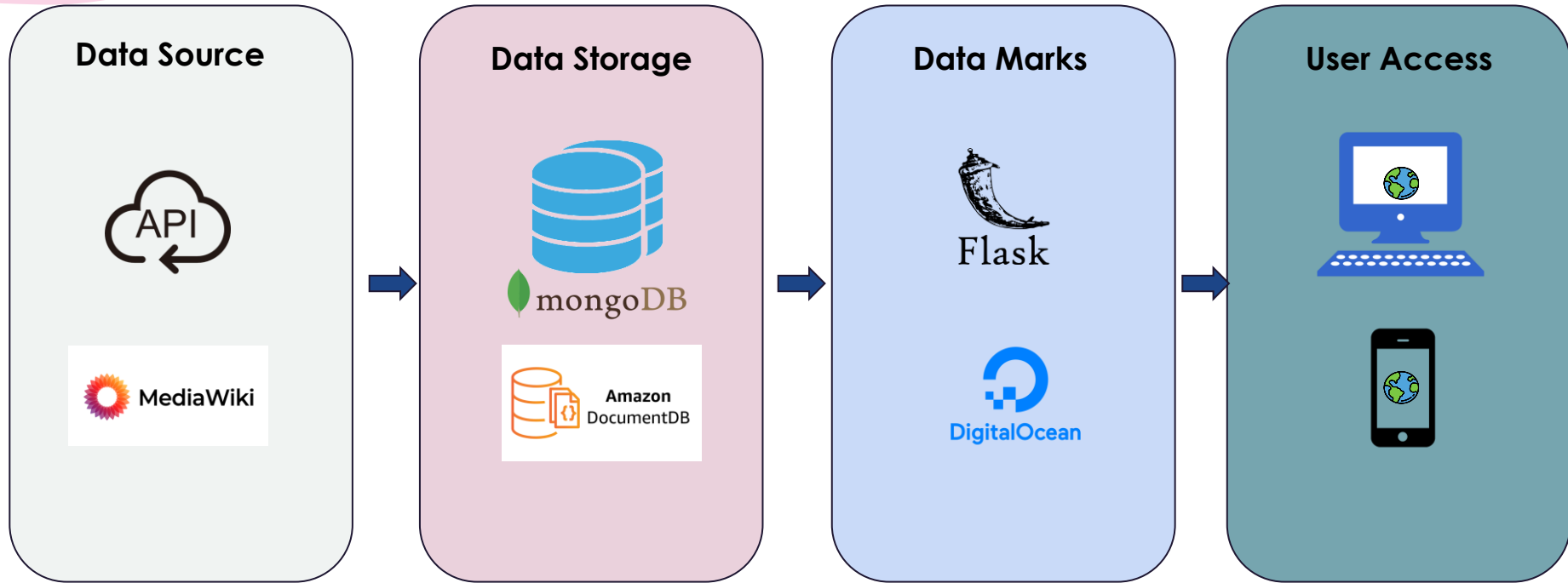


Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia. It utilizes the MediaWiki API that allows access to some wiki-features like authentication, page operations, search and parsing. Its advantages are its ease to use, simplicity, and automated requests.

MongoDB is chosen over Elasticsearch for its document store database model allowing schema-free organization of data. Such feature is suitable for NoSQL data requiring operations, though the MongoDB process may shut itself down in case of low system memory or high system resource utilization.

The development of the app would be executed using Flask, because of the following capabilities: full control of decisions to build the application given to the developer, little dependency to update, no need to lookout for security bugs.

Design



Data governance policies

Protecting Sensitive Personal Information of School Children & Teachers

1. Read-Only
2. Custom User Groups
3. Encrypt Columns
4. Consistent Account Audit/Clean Up

Data Access Policy

All registered teachers and students can access the application's data and information.

Data Usage Policy

Users should only access and use data for educational purposes; data sharing and data citation are encouraged.

Data Integrity Policy

Data should be complete, consistent, enduring and available throughout the life cycle of the application.



Cost Implications

Business Licensing Fee

The fee for accessing the Wikipedia API is \$21/month



Flask Application

For web hosting of our Flask application, we would use DigitalOcean's service for \$95/month.



Amazon
DocumentDB

Amazon Web Services

If our business is proven to be successful and we want to scale the application, we would use Amazon Web Service's DocumentDB (with MongoDB compatibility) for an extra \$60/month (depending on size of our data).

Assessment of success for the proposed metrics

Evaluation criteria		
Quantitative success		Qualitative success
Retrieval Time	User Engagement	Relevance of Searches
Application API Performance Metrics <ul style="list-style-type: none">- Request Per Minute- Latency	<ul style="list-style-type: none">- Retention Rate- Daily & Monthly Active Users- Average Session Length	Correctness <p>If the results match with the users' searching request.</p> <ul style="list-style-type: none">- Precision- Click through Rate
Mobile, Desktop App Performance Metrics <ul style="list-style-type: none">- App Load per Period- Average time to render screen	User Acquisition	Completeness <p>If the information displays well and answers users' requests.</p> <ul style="list-style-type: none">- Screen dimensions, font sizes, image and scripts display- Scoping
	<ul style="list-style-type: none">- Funnel conversion rates	

Conclusion & Future Recommendations

An app engaging students with interactive images and geographic features of the selected country is created.



Limitations (Business-side): (1) **Audience constraint.** The usage of the app relies on teacher's guidance for school kids to navigate themselves through the app (aged 10+). The audience is limited. (2) **Performance constraint.** For cybersecurity concerns, it's set to read-only by default.

Limitations (Technical-side): **Flexibility constraint.** Users could call the geographic features out only by calling the country name but not through other lookup tables.

Recommendations: Expand flexibility and functionality by adding more interactive elements like images, sounds, animation to engage users, and enlarging searching toolbars so that users can search by certain geographical features and have a list of countries or related facts returned.