Summary to Line Manager on SpaceOdyssey Project

Good afternoon,

This document contains a summary of the SpaceOdyssey automated radius updates project, including my concerns and potential risks.

Overview of the project

The ‘SpaceOdyssey - automated radius updates’ aims to automate the updating of radius and budget for Google Ads and Facebook Ads campaigns based on a CSV file received via email sent by the client weekly. The CSV file includes action column which specifies whether the ad should receive ‘boosted’ budget, ‘regular’ and ‘exclude’ which pauses the campaign all-together. We aim to visualise whether the sales for boosted ads increased by implementing a dashboard.

The key features development is separated into 3 phases, 1 refinement and testing phase, 1 deployment phase and lastly 1 maintenance and monitoring phase, so total of 6 phases. Firstly I’ll list these phases and provide a short summary and estimated timelines. A more detailed summary was provided to the client. A more details phase summary can be provided upon request. Secondly I’ll list any concerns and potential risks tied to this project.

Phases of the project

Phase 1: Email monitoring and CSV processing (Estimated time: 3 days)

1.1 Using Google Apps Script to monitor Gmail account for incoming emails with a specific subject and CSV file provided by the client. This script will run 3 times a week.

1.2 Passing the CSV data to AWS Lambda for processing.

Phase 2: AWS Implementation (Estimated time: 8.5 days)

2.1 Using S3 to store sales data.

2.2 Developing Lambda function to process CSV data and interacting with. Google/Facebook Ads APIs storing the sales data in S3 bucket.

2.3 Setting up API Gateway to exposing the Lambda function as a REST API.

2.4 Configuring IAM roles and policies such as least-priviledge for security and privacy.

Phase 3: Visualisation of sales data (Estimated time: 2 days)

3.1 Using Amazon QuickSight to visualise the sales data store in S3 bucket.

Phase 4: Final refinement, integration, and testing (Estimated time: 3 days)

Phase 5: Deployment of the project (Estimated time: 1 day)

Phase 6: Maintenance and monitoring using AWS CloudWatch

6.1 e.g. Opening a new Slack channel and have logging in the code that sends notifications there

Total estimated time: 17.5 days (plus 2-3 days buffer and client reviews)

Concerns and potential risks

1. **Integration complexity**
    - Coordination with Media Activation Team:
        - The project relies on access tokens and campain IDs from the media activation team for Google and Facebook Ads. Delays or issues with acquiring these credentials could potentially impact the timeline.
    - API Interaction:
        - Ensuring interaction with both Google and Facebook Ads APIs may present unforseen challenges. The APIs could undergo changes or have behaviour that hasn't been documented. It can also be challenging to integrate 3rd party APIs.

2. **Secutiry considerations**
    - IAM Policies:
        - Making sure least-privilage IAM policies and not impacting the funtionality when the services are integrated. Any misconfiguration could lead to insufficient permissions, which is less concerning, and concerning security vulnerabilities more challenging to spot.

3. **Email monitoring and data accuracy**
    - Email Script reliability:
        - Key projects functionality is based on Google Apps Script accurately detecting and processing incoming emails. Any issues with said script could result in failed updates or wrongly processed data.
    - Data accuracy and integrity:
        - The AWS Lambda must include edge case handling the CSV data to prevent processing errors.
    - Client seding email on time and in the correct format.

4. **Visualisation of sales data**
    - Setting up QuickSight:
        - QuickSight integrates well with other AWS services such as S3, although it requires a slight learning curve and planing to provide meaningful data visualisation.

5. **Junior Engineer involvement and resource management**
    - Potential delays:
        - Given the complexity of the projects solution there may be unforseen delays which could appear during integration or testing.
    - Learning curve:
        - Considering a junior engineer will be handling the project, there might be a steep learning curve, especially because of integration and security aspects. Support of a senior member of the team could ensure timely completion.

Final thoughts:
Allocating buffer time and extra time for customer reviews to the estimated 17.5 days could help with any timeline challenges presented throughout the project.
Thorough testing at each stage to catch any potential issues early could reduce post-depolyment problems.

By addressing these concerns proactively, I am confident we can mitigate these risks and ensure successful delivery of the project.

Best Regards,

Filip Mordych
Junior Python & Cloud Solutions Engineer