Confidential

Product Experimentation Two-Pager | Hi Pages Job Increase Job Leads 3 to 4 (I / II)

Situation: A product manager at hipages has an idea that might lead to higher lead utilisation by mandating 4 leads per job (usually 3). Asking for our help in performing an A/B test to categorise the change.

Context:

1. Homeowner posts a job against a category with basic details (description, location)
2. The Hi Pages platform then matches this to the most suitable tradies and those tradies can claim a lead and send a quote on a FIFO basis
3. Up to 3 chosen tradies send their quotes to the Homeowner (the experiment is to make this up to 4)
4. Our business model is based on charging tradies for the lead

Contact information

- James Maulana
- Creation Date: 18th Feb 2025
- Last edited: 18th Feb 2025

Objective: To design and execute an A/B test that evaluates whether increasing leads per job results in:

- Higher lead utilisation (more accepted leads, improved quote rates)
- Increased revenue from tradies
- No adverse impact on consumer experience

Hypothesis (H0): Mandating 4 leads per job will improve overall lead utilisation by providing tradies with additional opportunities to engage with consumers, thereby increasing the acceptance rate and improving platform revenue

Success Metrics & Expected Outcome

Primary:

- **[INCREASE]** Lead Acceptance Rate (percentage of leads accepted by tradies)
- **[INCREASE]** Conversion Rate (percentage of jobs leading to successful consumer-tradie interactions)

Secondary:

- **[INCREASE]** Average Revenue per Job
- **[INCREASE]** Consumer Engagement Metrics (e.g., time to first quote, Acceptance rate)

Experiment Design

Experimental Groups:

- **Control Group:** Jobs receive the current standard of 3 leads
- **Test Group:** Jobs receive 4 leads

// As control group and test group for the job_post_event would show as 1–3 or 1–4 do not need to use experiment_group as flag for back-end analysis //

Randomisation:

- Randomly assign incoming job postings to either the Control or Test group (e.g. random number generator for job assignment)
- For this experiment 50/50 split best vs. skewed split (80/20) as we're dealing with a non-high stake feature
- 50/50 gives faster learning and cleaner comparison

Tools:

- A/B testing Tools: **Optimizely**, **Google Optimize**, etc. or inhouse solution with feature flags (tag accounts) as this is a server-side experiment
- Visualisation Tools: **Tableau**, **Looker**, etc. with livestream to database (e.g. BigQuery)

Data Structure & Event Taxonomy (example):

// Standardise timestamps across all events to make downstream analysis easier//

Job_Post_Event:

- job_id, category, location, job_open_leads (1-3/4), posted_timestamp, experiment_group (Control, Test)

Lead_Assignment_Event:

- job_id, tradie_id, lead_position (1-3 or 1-4), assigned_timestamp

Lead_Viewed_Claimed

- job_id, tradie_id, viewed_timestamp, claimed_timestamp

Lead_Acceptance_Event:

- job_id, tradie_id, accepted (boolean), response_time, acceptance_timestamp

Job_Conversion_Event:

- job_id, consumer_id, tradie_id, conversion_timestamp

Job_Completion_Event:

- job_id, completion_status (e.g., completed, cancelled, in dispute), completion_timestamp

Product Experimentation Two-Pager | Hi Pages Job Increase Job Leads 3 to 4 (II / II)

Technical Execution & Measurement
// Pre-Experiment Preparatory Work //

Technical execution
1. Modify backend routing logic to allow either 3 or 4 lead assignments based on the randomisation flag (*see job_post_event schema experiment_group flag*) using random number generator
2. Ensure the lead assignment logic checks the randomization flag before deciding the number of leads (3 vs. 4)
3. Tag every relevant event (Job_Post, Lead_Assignment, etc.) with experiment_group (Control or Test) to facilitate downstream analysis
4. Stream events (e.g., using tools like **Kafka** or **AWS Kinesis**) into a data warehouse project (e.g., **BigQuery**) for near real-time analysis
5. Validate event payloads (job_id, experiment_group, etc.) upon ingestion to maintain data quality
6. Test company A/B testing platform or inhouse solution with feature flags (tag accounts) are working
7. Test streaming into chosen
8. Use visualisation tools with live stream (frequent ingestion) to monitor acceptance, conversion, and revenue metrics

// Note: Optimizely does not have advanced Bayesian analysis like Beta-Bernoulli estimations etc., therefore require database to complete some manual analysis//

Measurement
1. Confirm that each job is clearly labelled with experiment_group (Control vs. Test).
2. Use a **50/50** split in A/B tool (e.g. Optimizely to ensure each variant has adequate sample size
3. **Check** randomisation across user segments (e.g., categories, regions) through a test, and then again on the day to confirm no unintentional bias
4. Gather historical acceptance and conversion rates to compare test results.
5. Document known exogenous lifts (promotions, external events) to avoid misattributing spikes in acceptance
6. Avoid major holidays or events (e.g., Christmas, sports finals) that could skew normal user behaviour
7. Establish a minimum run time (e.g., 2–4 weeks) or until the test reaches the required sample size for statistical power

Tracking, Analysis and Education

Tracking:
- Create real-time or near real-time dashboards (using **Tableau**, **Looker**) to track acceptance rate, conversion rate, revenue, and lead coverage for Control vs. Test
- Include segmentation filters (category, region) to spot anomalies or segment-specific trends
- Store event logs in a data warehouse to complete more robust analysis in SQL and Python later; that experimentation platform (e.g. Mixpanel) doesn't offer

Analysis:
1. Frequentist & Bayesian Approaches
- **Frequentist** (e.g., t-tests, z-tests) to compute p-values and confidence intervals for acceptance or conversion rate differences between Control and Test
- **Bayesian** (e.g., Beta-Bernoulli) to estimate posterior distributions of acceptance rates, providing probabilities of one variant outperforming the other

2. Confounding Variables & Effect Sizes
- Account for category differences, time-of-day patterns, or location bias (e.g., some categories more prevalent in the Test group)
- Emphasize **effect size** (how big the improvement is in absolute and percentage terms) alongside statistical significance

3. Stopping Criteria
- Use a **sample size calculator** or power analysis to decide when we have enough data for confident conclusions (e.g., detect a 2% absolute lift with 80% power)
- **Bayesian** sequential analysis would allow us continuous monitoring and potentially stop the test early if results are clear

Education:
Provide a short internal workshop or "dry run" session with Product team that covers:
- Methodology choice
- Key differences between frequentist p-values and Bayesian posterior probabilities
- How to interpret each method's outputs (confidence intervals vs. credible intervals)
- When we can stop based on effect size

Next Steps:
1. **Deploy Pilot:** Roll out the A/B test to a controlled percentage of job postings for a defined period (e.g., 2-4 weeks)
2. **Monitor & Analyse:** Continuously monitor performance metrics; if results are statistically significant and positive, prepare to scale the feature to all users
3. **Recommendation for Scaling:** If experiment confirms that 4 leads per job significantly boost lead utilisation without negative side effects, proceed to a full rollout with targeted regional adjustments and ongoing monitoring

FAQ – For Team Education

1. Why is tracking important?
    - The main thing we want to test for is the impact of the feature release, and how the feature release effects user behaviour. The best way to do this in a robust manner is to do a side-by-side comparison with a control group.
2. Can you do this full analysis with Optimizely?
    - Optimizely provides basic tracking for A/B testing and offers some basic statistical analysis, but it does not allow for more advanced statistical methods like Bayesian analysis. Storing data in a database with actual tags (a new column) or by assigning based on uuid, hash, or another method, would allow for more advanced analysis.
3. What is the frequentist approach and why does it matter?
    - **P-value:** A low p-value (typically below 0.05) indicates statistical significance. This suggests that the observed difference between the control and test groups is unlikely to be due to chance
    - **Confidence Intervals:** If the confidence interval for the difference in conversion rates does not include zero, it suggests a statistically significant difference
    - Typicall you would stop this experiment only once you've collected at least n observations per group (or enough to detect your effect), you check if your difference is statistically significant at (typically) $\alpha=0.05$. If so, you can stop the test. This is typically a one-off calculation and can be run using an online calculator
4. What is the Bayesian approach and why does it matter?
    - Calculate the probability that one variation outperforms the other based on the posterior distributions. A high probability (e.g., > 95%) indicates a strong likelihood that one variation is superior
    - As it's continuous we can determine for ourselves when to stop the experiment, e.g. "Stop when there's a 95% chance that Test is better than Control by at least 2%.". If that threshold is reached, we can end the experiment early; if it remains unclear we would continue to gather data