

Blofin Announcement Feed Integration Plan

Summary for Blofin Team

- **Objective:** Integrate Blofin's **Announcement Center** updates (new listings, promotions, etc.) and **price pump alerts** into our application's newsfeed in real-time, to drive user engagement and trading volume via Miles Deutscher's referral.
- **Data Sources:** We will use Blofin's **public APIs** (Help Center API for announcements, and Market Data API for prices) to **pull announcements and market alerts** as soon as they go live. All API usage will be **read-only** and low-frequency (polling approximately every 1 minute) to avoid heavy load.
- **What Will Be Shown:** New **spot listings**, **perpetual contract listings**, and major **news/promotions** will appear in our newsfeed with the announcement title and a **"Trade on Blofin"** link (embedded with our referral code). Additionally, alerts for tokens **surging in price** beyond a certain threshold will be displayed, mirroring Blofin's in-app popups.
- **Technical Approach:** Our backend will periodically query Blofin's help center API for new articles in the Announcement Center (e.g. *Spot Listing*, *Futures Listing* sections) ¹ ² . It will also monitor Blofin's market data (either via REST or WebSocket) for rapid price movements. No data is written or modified on Blofin's side – we only **retrieve public information**.
- **Benefit:** This integration relays Blofin's latest updates instantly to our user base, encouraging timely trading activity on Blofin (with referral tracking), without impacting Blofin's infrastructure (minimal API calls). We seek Blofin's approval for this plan and any guidance on best practices or dedicated endpoints for announcements/pump alerts if available.

Data Sources: Blofin Announcement Center

Blofin maintains an **Announcement Center** (on their support site) organized into multiple categories for different types of updates ³ . Key categories include:

- **Latest News:** General platform news (feature launches, partnerships, etc.).
- **Latest Promotions:** Marketing events and trading competitions (e.g. fee discounts, prize pools) ⁴ .
- **Spot Listing:** Announcements of new coins/tokens listed on spot markets ¹ .
- **Futures Listing:** Announcements of new perpetual contracts/futures listed ² .
- (Other categories like *Delisting*, *Trading Updates*, *Systematic Service*, etc. also exist, but our integration will primarily focus on the above, which directly drive trading activity.)

Each announcement is published as an article in the help center. For example, under **Spot Listing** you will see entries like *"BloFin Will List ESPORTS/USDT for Spot Trading (2025-07-19)"*, *"Upcoming Listing: BloFin to List TRI for Spot Trading"*, etc. ⁵ . Under **Futures Listing**, you have entries like *"ERAUSDT Perpetual Contract Listing: Enjoy 0 Maker Fees..."*, and so on ⁶ . These titles contain the core info (asset name and listing date or promotion details). We will leverage these to inform users.

Accessing Announcements via API: Blofin's support site is powered by Zendesk Guide, which provides a **public Help Center API** for knowledge base articles. We can retrieve announcements in JSON format by querying the appropriate endpoints. For instance, Zendesk's API allows listing all articles in a section with a GET request to a URL like:

```
https://<subdomain>.zendesk.com/api/v2/help_center/<locale>/sections/  
{section_id}/articles.json
```

In Blofin's case, the subdomain is `blofin` (with custom domain support via support.blofin.com). We have identified the section IDs for relevant categories (e.g., Spot Listing, Futures Listing, etc.) and will use those. For example, a request to the Spot Listing section might look like:

```
GET https://blofin.zendesk.com/api/v2/help_center/en-us/sections/11892707214991/  
articles.json
```

This returns a JSON object containing an array of articles with fields like `id`, `title`, `html_url`, `created_at`, etc. (for all articles in that section) ⁷. Our system will parse this to find new announcements. The Help Center API is public-read, so no authentication is required for retrieving public articles ⁸, which keeps integration simple and **non-invasive**. (We will implement proper paging or use the provided pagination links in the JSON to ensure we catch all new articles ⁹.)

Implementation Plan

1. Polling for New Announcements:

We will set up a scheduled task (running **every minute**) that queries Blofin's announcement sections via the API. Specifically:

1. **Fetch latest articles** from the *Spot Listing* section (for new spot market tokens) and the *Futures Listing* section (for new perpetual contracts). We may also poll *Latest News* and *Latest Promotions* sections to catch any other relevant updates (e.g. major news like new features or promotional campaigns). This can be done with a lightweight HTTP GET request returning JSON data ⁷.
2. **Compare with last seen article** – We will track the ID or publication timestamp of the last announcement we've processed for each section. If the API returns an article that is not in our recorded list (i.e., a newly published announcement), our system will recognize it as a new event.
3. **Record and Display** – For each new announcement, we will format a newsfeed entry. This entry will include the announcement **title** (e.g. *"BloFin Will List VELVET/USDT for Spot Trading (2025-07-10)"* ¹⁰) and a direct **link** to the full announcement on Blofin's site. We will also provide a **"Trade Now on Blofin"** button or link. The trade link will be constructed based on the content of the announcement: for a spot listing, we'll use Blofin's spot trading URL for that pair; for a futures listing, the futures trading URL. For example:
 - Spot pair XYZ/USDT → link to `https://blofin.com/spot/XYZ-USDT` (Blofin's spot trading interface for that coin) ¹¹.
 - Perpetual contract XYZUSDT → link to `https://blofin.com/futures/XYZ-USDT` for the USDT-margined perpetual ¹².

These links will have our referral embedded on the backend (e.g., ensuring the user sign-up or activity counts towards *Miles Deutscher's* referral code). If Blofin has a specific format for referral tracking in URLs, we will use that (for instance, appending the invite code or using the invite link if the user is not yet registered).

Technical note: The announcement JSON from the API includes a direct `html_url` to the article on Blofin's support site, which we will use as the "source" link if the user wants to read details. But the **"Trade" link will go straight to Blofin's trading page** for convenience. We'll parse the ticker symbol from the announcement title (which consistently contains the token symbol and market) to build the trading URL. For example, title "Will List **VELVET/USDT**..." indicates symbol **VELVET** with USDT pairing – our system maps that to `blofin.com/spot/VELVET-USDT`. Similarly, "**PUMPUSDT** on BloFin Perpetual" indicates a futures contract **PUMP-USDT** ¹³, mapped to `blofin.com/futures/PUMP-USDT`.

2. Handling New Listing Announcements:

Whenever a new listing announcement is detected, it will appear in the feed as something like:

- " *BloFin Lists VELVET/USDT on Spot – deposits open 12:00 UTC, trading starts 12:30 UTC* ¹⁴ . **Trade on BloFin.**"

This gives users a quick heads-up with essential info (we can even parse the schedule from the announcement body – e.g. trading start time – to include it). The **Trade on BloFin** link would send the user to Blofin's trading interface for that asset. By clicking it, users can immediately start trading the newly listed asset (and if they're new to Blofin, the referral code ensures they sign up via *Miles Deutscher's* invite).

(If an announcement is an "Upcoming Listing" (pre-announcement of a future listing), we will still show it – possibly labeled as "Upcoming". The trade link in that case could direct to a placeholder or simply Blofin's markets page, since trading isn't live yet, along with a note of the go-live time.)

3. Handling Promotions and News:

Announcements in *Latest Promotions* (like trading competitions or fee discounts) will also be pulled in. For example, "*Pump & Win: 20,000 USDT Bonus Challenge*" might be an event where traders who capitalize on token pumps get rewards ¹⁵. These will be displayed with a brief description and link to details. The integration of promotions encourages users to participate (again increasing Blofin volume). Similarly, any major *Latest News* (e.g. "*Buy Crypto with Apple Pay — Now Available on BloFin!*") can be shown to keep users informed ¹⁶.

4. Pump Alert Integration (Real-Time Token Pump Popups):

Blofin's app has a feature where a popup alerts users if a token's price pumps by a certain percentage within a short time. We aim to replicate these alerts in our newsfeed. Since there is no explicit "announcement" for these, we will generate alerts by monitoring market data:

- We will utilize Blofin's **Market Data API** to get price updates. Blofin provides a public endpoint for ticker data (latest price, 24h change, volume, etc.) ¹⁷ ¹⁸. For example, `GET /api/v1/market/tickers` returns an array of all trading pairs with their last price and 24h stats ¹⁹. We can call this periodically (e.g. every 1 minute alongside announcements polling) to compute price change over the last interval.

- For more immediate detection, we could use Blofin's **WebSocket tickers channel**, which streams live price updates with up to 1-second frequency without requiring auth ²⁰. This would allow us to catch sudden movements in real-time. Depending on complexity, we might start with REST polling and later upgrade to the WebSocket for even faster alerts.
- We will define a **threshold** for a "pump" – for instance, **price increase >5% within 5 minutes** (this exact threshold can be adjusted to match Blofin's criteria if we get that info from the team). Our system will track short-term price changes for each asset: if any coin's price jumps rapidly beyond the threshold, we create a newsfeed alert like: "📈 *TOKEN surged +X% in the last Y minutes on BloFin! Trade now.*". This alert would include a trade link to that token's market on Blofin.
- These pump alerts will be generated in a way that mirrors Blofin's own in-app alerts, so our users get the same timely info. We will ensure the frequency of checks (every minute or via stream) is efficient. Using the ticker API is not intensive – e.g., one API call for all tickers yields current prices and 24h change ²¹, which is a lightweight JSON that we can quickly scan for large changes.

5. Frequency & Efficiency:

All checks are done roughly every 60 seconds, which strikes a balance between speed and API usage. One minute polling for announcements is more than sufficient (new announcements aren't more frequent than a few per day). Even if we check four sections, that's 4 requests per minute. The Help Center API can easily handle this and it's read-only. For price data, if using REST, one request per minute to `/market/tickers` (or a few specific tickers of interest) is also minimal. If using WebSocket for tickers, it's a single persistent connection streaming data, which is efficient and avoids repeated HTTP calls. In both cases, the load on Blofin's servers is negligible. Our integration is **read-only** – no account or trading APIs are invoked – and we respect API rate limits and best practices.

6. Example Workflow:

For clarity, here's a hypothetical sequence when Blofin releases a new announcement:

- **Time 0:** Blofin publishes "BloFin Will List XYZ/USDT for Spot Trading (2025-08-01)" on the Announcement Center.
- **Time +30s:** Within the next minute, our service polls the Spot Listing section and retrieves this new article (with title, timestamp, etc.) ¹⁰ ¹⁴. The system sees a new title "XYZ/USDT for Spot Trading" that wasn't there before.
- **Time +31s:** Our parser extracts "XYZ" as the token symbol and identifies it as a spot listing. It creates a newsfeed entry: " *BloFin will list XYZ/USDT for spot trading on Aug 1, 2025. Trade on Blofin.*" The "Trade on Blofin" link is set to `https://blofin.com/spot/XYZ-USDT` (and we ensure if the user hasn't registered, the referral invite is applied in the flow).
- **Time +32s:** The newsfeed in our app/website now shows this item to all users. Any user clicking it can read details (we can link the announcement title to the full Blofin announcement page) or jump straight to trading on Blofin.
- **Time +5min:** Suppose the price of XYZ token starts trading and surges 10% in the first 5 minutes. Our system, through the ticker feed, notices this rapid jump (10% > 5% threshold). We then display: "📈 *XYZ price is up 10% in 5 minutes on BloFin!*". This alert again has a direct trading link. Users get immediate feedback that XYZ is hot, prompting them to trade if interested.
- This way, our users stay **fully synced with Blofin's updates** – from official listings to real-time market movers – nearly as fast as Blofin's own platform.

Considerations and Next Steps

- **API Stability:** We assume Blofin's support center API endpoints (via Zendesk) remain accessible. In case Blofin has or can provide a dedicated feed (e.g. an official RSS or webhook for announcements), we can switch to that for even more efficiency. We will coordinate with Blofin's tech team on the best endpoints to use.
- **Threshold Tuning:** For pump alerts, if Blofin can share the exact parameters they use for triggering the in-app popup (e.g. "X% rise in Y minutes"), we will align our alert logic to match. Otherwise, we'll use a reasonable approximation as described.
- **Referral Link Integration:** We will work with Blofin to ensure that the referral tracking is correctly implemented. Our understanding is that using the invite link (`blofin.com/invite/MilesDeutscher`) for new sign-ups is required; we will embed this where appropriate (likely when a user without a Blofin session clicks to trade, they'll be routed through the invite link or code application process). For already registered users, the direct trading links will simply open the Blofin interface.
- **Testing:** Before going live, we'll test the integration with a dry run – e.g., fetch recent announcements (like the Velvet token listing) and verify our system captures it and forms the correct links. We'll also simulate a price pump scenario with historical data to ensure the alert triggers as expected.

By implementing the above plan, our application will **echo Blofin's announcements and market excitement in real-time**, creating a seamless experience for users to follow news and immediately act on it. This should benefit Blofin through increased engagement and trading volume via our referral partnership, while providing value to our users. We kindly request Blofin's approval of this approach and any additional guidance (such as rate limits, or existing feeds for announcements/pumps) to ensure a smooth integration.

Sources: Blofin Help Center – Announcement categories and examples ¹ ² ; Blofin API Documentation – public market data (tickers) endpoint ¹⁷ ¹⁹ ; Zendesk Developer Docs – help center articles API usage ⁷ ⁹ .

¹ ² ³ ⁴ ¹⁵ Announcement Center – Help Center

<https://support.blofin.com/hc/en-us/categories/6149031297551-Announcement-Center>

⁵ Spot Listing – Help Center

<https://support.blofin.com/hc/en-us/sections/11892707214991-Spot-Listing>

⁶ ¹³ Futures Listing – Help Center

<https://support.blofin.com/hc/en-us/sections/6200516139919-Futures-Listing>

⁷ ⁸ ⁹ How to get all Zendesk help articles using their API - Stack Overflow

<https://stackoverflow.com/questions/76508615/how-to-get-all-zendesk-help-articles-using-their-api>

¹⁰ ¹⁴ BloFin Will List VELVET/USDT for Spot Trading (2025-07-10) – Help Center

<https://support.blofin.com/hc/en-us/articles/13210529944975-BloFin-Will-List-VELVET-USDT-for-Spot-Trading-2025-07-10>

¹¹ 0.00 | ETH/USDT | Trade ETH/USDT on BloFin

<https://blofin.com/spot/ETH-USDT>

12 BTC/USDT | Trade BTC/USDT, Share Your Position Profits & Loss (PnL) with Friends

<https://blofin.com/futures/BTC-USDT>

16 BloFin Adds Apple Pay Support, Enhancing a Seamless Crypto

<https://www.globenewswire.com/news-release/2025/07/16/3116717/0/en/BloFin-Adds-Apple-Pay-Support-Enhancing-a-Seamless-Crypto-Buying-Experience-for-Traders.html>

17 18 19 20 21 BloFin API guide | BloFin API Documents

<https://docs.blofin.com/index.html>