KVI MESSENGER CHAIN Whitepaper

Abstract

This whitepaper aims to explain the objectives and the intended approach for the KVI Messenger Chain – an innovative block-chain solution for mobile-based digital content and community management

Dr Chung Ling Soo, KVI

Floor 2, Daedong Tower

Dosan-daero, Gangnamgu, Seoul TEL 02-541-6383

Table of Contents

Background

This initiative is being undertaken by Redision, who are a licensing partner for Telkomsel, Indonesia's state-run news agency and by local corporation TAP Mitra Global.

On December 8th, a business agreement was signed that establishes an agreement with Telkomsel to carry out comprehensive work on the development and supply of a blockchainbased system to support HAI (tentative name) – a user-community platform and digital content supply/management system.

As part of the agreement, TEAM KVI have secured exclusive operator status for any related work in South Korea once the platform has been rolled out in Indonesia. This represents a significant agreement for future growth and expansion.

This WhitePaper outlines Team KVI's proposed approach to the design and delivery of this initiative.

About Telkomsel

Before we proceed with the technical content, it is important to understand who the thirdparties are in the project, and the scope of this undertaking, which is significant:

Telkomsel is a subsidiary of Telcom, who's customers include more than half of Indonesia's total mobile service subscribers. In addition to this, Telkomsel provide technical support for telecommunication services not only to Indonesia but also to about 10 further regions including Southeast Asia and South Africa.

In total, Telkomsel has about 165 million users and is the largest telecommunications operator in Indonesia and Southeast Asia.

Telkomsel's mobile payment business is owned by Indonesia's Redision, and new businesses are being considered according to local conditions.

As a result, the KVI-Chain will automatically be exposed to a very large user-base and should therefore be designed to scale accordingly.

There are two major use-cases for the KVI Messenger Chain initially – we shall explain both of these in detail next, to give an idea of the areas in which KVI can operate:

KVI Use-case 1: HAI and KVI

HAI is on track for launch very soon – development is currently approximately 80% competed. As part of this initial release, there will be a user-centered differentiated localization service that supports PC use as well as various smartphones such as iPhone, Android, and Windows Phone in the first half of 2020.

With this first release, users will be able to share and communicate multimedia such as pictures and videos through phone number authentication without having to sign up and sign in separately.

The real innovation, however, lies in how the users will be able to communicate: the language transmitted by the other party during a bi-lingual dialogue will be supported by a real-time automatic interpretation service. Further than this, the users will be able to apply enhancements to their dialogue via character voice filters, video filters and 'active emoticons' to deliver uniquely crafted, detailed emotional expression giving a unique user experience!

And if users are allowed to search the surrounding location, communication between users within a 2-kilometer radius will be supported, providing the facility to host 'local communities' and based on the users' location, the company will provide a function to search for restaurants,

pharmacies, gas stations, and banks within the range of the search that can be set to enhance the users interaction with their local surroundings.

The KVI Messenger Chain provides a blockchain-based infrastructure to support HAI, including a token-based financial model to allow flow and management of content through the system in such a way as to encourage contributors and reward according to utilisation.

Facilitated by the launch of HAI, TEAM KVI has agreed to distribute various high-quality content such as games, K-pop, hallyu dramas, excellent movie content, educational programmes, animation, music (soribada), performances, and general broadcasting, along with China's “Mango TV”, which is a platform for Korean game developers and Asian content producers to deliver further added value. Through these various functions, KVI will be activated and become central to the ecosystem, in order to create a healthier economic structure and provide consumers with optimal convenience.

TEAM KVI will also actively seek related businesses such as idol incubators for finding emerging stars in Southeast Asian and global markets who dream of becoming K-pop stars and others based on the HAI service base through participation in "God of Event," an entertainment O2O platform that links events with K-pop-focused hallyu stars in Korea. The KVI platform facilitates the onboarding and engagement of these rising stars and encourages them on to HAI, giving them an immediate enormous potential audience and therefore making their involvement highly lucrative for all parties.

KVI Use-case 2: VINAKON and KVI

Vietnam, a country currently very optimistic about achieving this year's economic growth target of 6.8 percent, has reached its highest level of economic success in the past nine years, making the possibility of even further growth very likely.

Vietnam, along with the U.S., China, Russia, South Korea and Japan, is one of the leading countries with high levels of investment and transaction frequency in cryptocurrency, with an estimated volume of USD 22,000 Anpak per day trading volume, making the country one of the highest crypto-trading regions in the world.

The Vietnamese government recently approved the establishment of VINAKON, the first global exchange that can deposit and withdraw money in VND. The VINAKON exchange, which is expected to operate stably with its strategic alliance with MB Bank, the fifth largest banking industry in Vietnam, is attracting much attention as it expects a trading volume and transaction fee revenue far beyond that the existing P2P exchange.

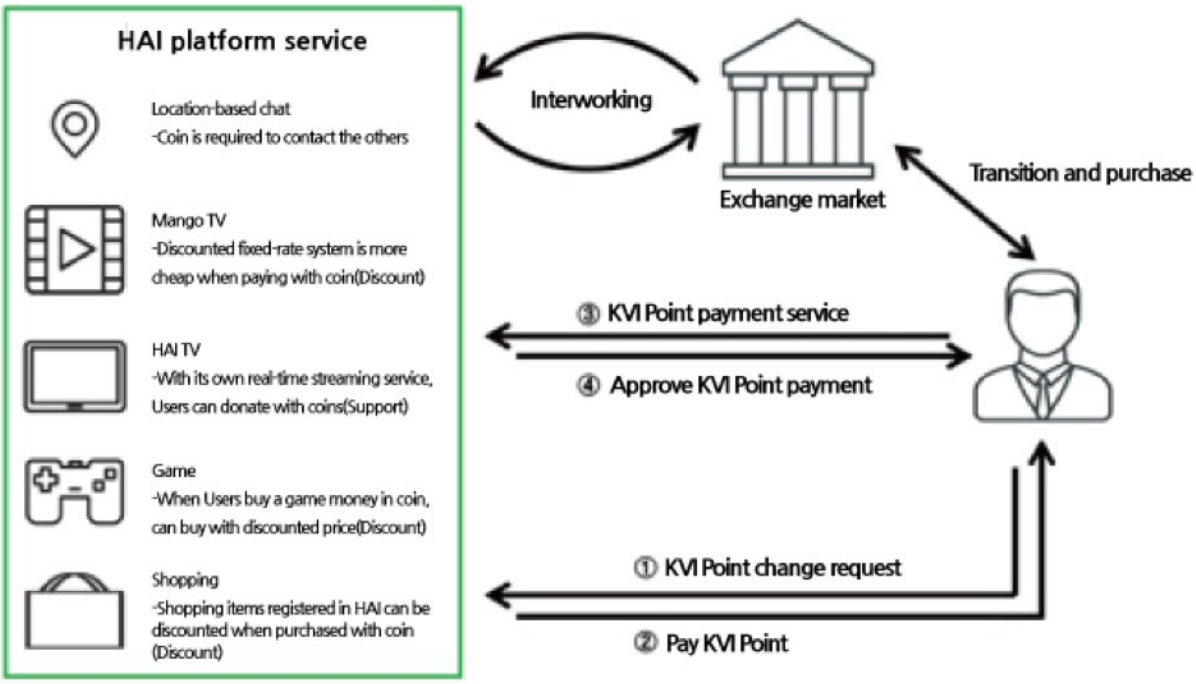
TEAM KVI is currently discussing a strategic partnership with the VINAKON Exchange, which plans to issue a project coin to invest in Vietnam's government-led social overhead capital project. This company plans to push for more aggressive business expansion in Southeast Asia.

This partnership will involve the KVI Messenger Chain to be used as the fundamental payment system and communications platform for the VINAKON exchange – again immediately

introducing a massive volume of users onto the platform and immediately elevating the KVI system to levels of extremely high utilisation.

The KVI Messenger Chain

Taking the HAI and the Exchange use-cases together, this is how the KVI chain will integrate with them to provide a coherent ecosystem that engages both parties, but also which utilises the Exchange to enable the HAI platform service itself:



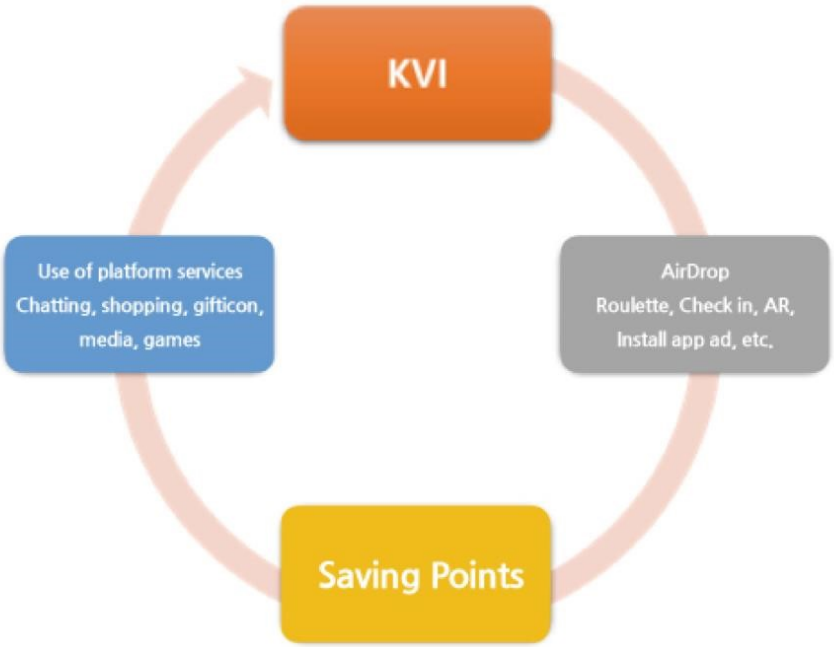
As you can see, the messenger-chain enables the entire economic system for HAI as well as being the glue for the communications between HAI and third-parties. Transactions will involve:

- Purchasing of HAI products and services
- Engagement of talent
- Payment/Reward for utilisation of talent-driven-content
- Microtransactions for gaming content
- Payment/Reward for utilisation of games on-platform

These are just a few key touch-points; KVI will be utilised throughout any future offerings HAI chooses to include in their platform.

From a users' perspective, using the KVI token as the platform currency will mean transaction overheads are kept to a minimum and are extremely fast: users get a highly performant system for a fraction of overhead charges incurred via traditional fiat-driven systems.

In addition to this fundamental financial system, KVI also introduces a token-driven 'Reward Points' system giving users the ability to earn points, which can then be used to purchase further products and services, and to promote various new channels, games and other content.



This effectively provides a self-driving promotional and marketing system within the platform, meaning consumers of KVI do not need to implement such systems themselves, saving significant time and effort.

Messaging System

Finally, at the heart of KVI lies the actual messaging system. The purpose of this is to allow consumers to communicate with one another seamlessly, both with content (as per the HAI use-case) or for transactions (the VINAKON use-case) or for simply facilitating and negotiating the movement of resources/objects between system.

To explain this more clearly, we need to understand the proposed architecture:

KVI Messenger Chain – Architecture

KVI MQ: The Messaging System

We have chosen to implement the KVI messaging system based on the RabbitMQ distributed message queue system. The RabbitMQ system is traditionally of distributed design because it is usually run as a cluster of nodes where queues are spread across the nodes and optionally replicated for fault tolerance and high availability. It natively implements AMQP 0.9.1 and offers other protocols such as AMQP 1.0, STOMP, MQTT and HTTP via plug-ins.

This is a perfect model for Blockchain, which is by its very nature both distributed and nodeoriented, hence the reason for adopting this approach.

Our messaging system shall take both a classic and a novel take on messaging. Classic in the sense that it is oriented around message queues, and novel in its highly flexible routing capability. It is this routing capability that is its killer feature. Building a fast, scalable, reliable distributed messaging system is an achievement in itself, but the message routing functionality is what makes it truly stand out among the myriad of messaging technologies out there.

RuleManagers and Queues

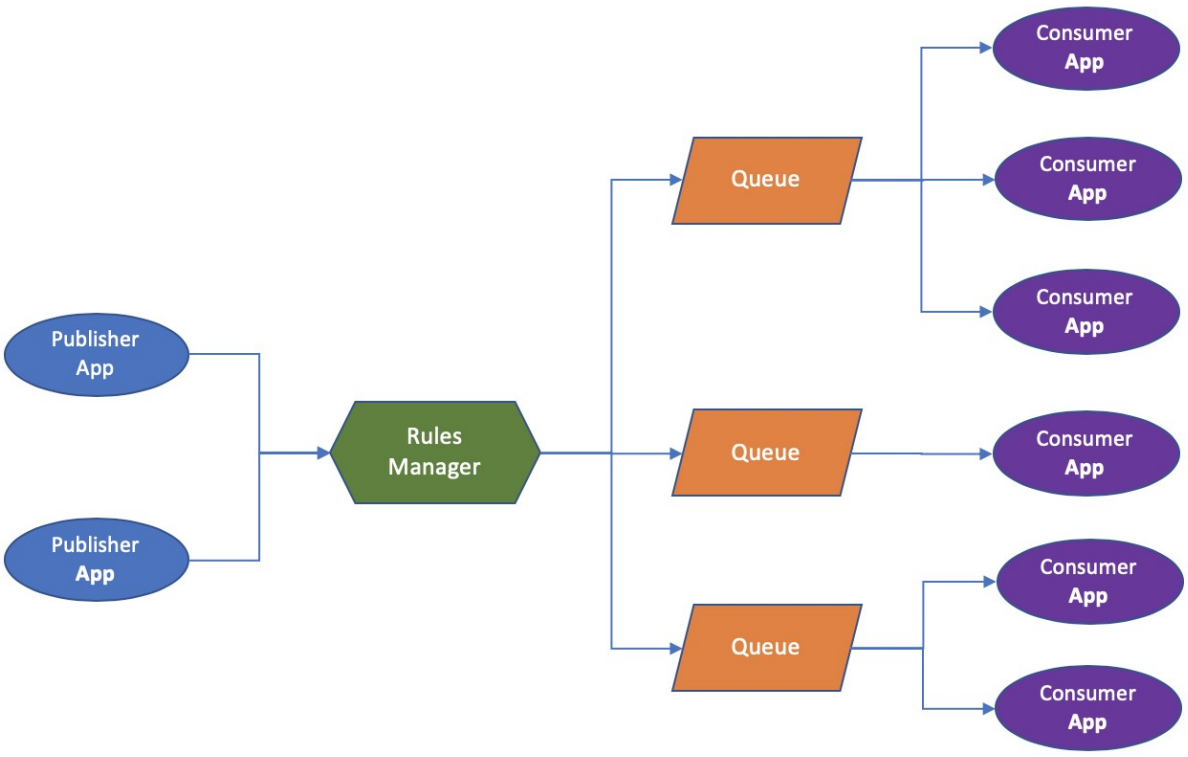
The super simplified overview:

- Publishers send messages to RuleManagers
- RuleManagers route messages to queues and other RuleManagers
- KVI sends acknowledgements to publishers on message receipt
- Consumers maintain persistent TCP connections with KVI and declare which queue(s) they consume
- KVI pushes messages to consumers
- Consumers send acknowledgements of success/failure
- Messages are removed from queues once consumed successfully

RuleManagers are essentially a set of routing rules. You send a message to a channel process, it uses the routing rules (RuleManagers) to decide where to send the message on to.

Hidden in that list are a huge number of decisions that developers and admins should take to get the delivery guarantees they want, performance characteristics etc, all of which we'll cover in later sections of this series.

Our RulesManagers need to be able to handle multiple messages and route them to multiple queues correctly. This diagram illustrates a typical example of how the messaging would flow:



Messages are delivered in order of their arrival to the queue (that is the definition of a queue after all). This does not guarantee the completion of message processing matches that exact same order when you have competing consumers. This is no fault of KVI MQ but a fundamental reality of processing an ordered set of messages in parallel. This problem can be resolved by using the Consistent Hashing Exchange.

Push and Consumer prefetch

KVI MQ pushes messages to consumers in a stream. There is a Pull API but it has terrible performance as each message requires a request/response round-trip.

Push-based systems can overwhelm consumers if messages arrive at the queue faster than the consumers can process them. So to avoid this each consumer can configure a prefetch limit (also known as a QoS limit). This basically is the number of unacknowledged messages that a consumer can have at any one time. This acts as a safety cut-off switch for when the consumer starts to fall behind.

Why push and not pull? First of all it is great for low latency. Secondly, ideally when we have competing consumers of a single queue we want to distribute load evenly between them. If each consumer pulls messages then depending on how many they pull the distribution of work can get pretty uneven. The more uneven the distribution of messages the more latency and the

further the loss of message ordering at processing time. For that reason KVI MQ's Pull API only allows to pull one message at a time, but that seriously impacts performance. These factors make KVI MQ lean towards a push mechanism. This is one of the scaling limitations of KVI MQ. It is ameliorated by being able to group acknowledgements together.

Routing

Exchanges are basically routing rules for messages. In order for a message to travel from a publisher channel process to a queue or other exchange, a binding is needed. Different exchanges require different bindings. There are many types of exchange and associated bindings:

Fanout. Routes to all queues and exchanges that have a binding to the exchange. The standard pub sub model.

Direct. Routes messages based on a Routing Key that the message carries with it, set by the publisher. A routing key is a short string. Direct exchanges route messages to queues/exchanges that have a Binding Key that exactly matches the routing key.

Topic. Routes messages based on a routing key, but allows wildcard matching.

Header. RabbitMQ allows for custom headers to be added to messages. Header exchanges route messages according to those header values. Each binding includes exact match header values. Multiple values can be added to a binding with ANY or ALL values required to match.

Consistent Hashing. This is an exchange that hashes either the routing key or a message header and routes to one queue only. This is useful when you need processing order guarantees with scaled out consumers.

Dead Letter Managers

We can configure queues to send messages to a RulesManager under the following conditions:

- Queue exceeds the configured number of messages.
- Queue exceeds the configured number of bytes.
- Message Time To Live (TTL) expired. The publisher can set the lifetime of the message and also the queue can have a message TTL. Whichever is shorter applies.

We create a queue that has a binding to the dead letter manager and these messages get stored there until action is taken.

This concludes the architecture for the KVI MQ; essentially modelled on RabbitMQ and an ideal candidate for blockchain implementation, we believe this is a perfect fit for KVI generally.

The KVI Token

The KVI token will initially be Ethereum (ERC-20) based initially, since this allows KVI to easily implement complex smart-contracts easily. It is expected that KVI will move to its own native coin in the future, however the priority is to deliver core functionality as quickly as possible, to facilitate HAI and VINAKON, since both of these projects have relatively aggressive deadlines.

However, we are designing KVI as though it would launch as Native, since this approach means we build the system properly and do not have to rebuild compromised designs based on the short-term solution. For this reason, we include transaction-based design approaches as well as the fundamental security implementation:

Transactional performance

KVI will implement a Merkle-tree based architecture for its transaction management, since this works extremely well across a decentralised system, is very fast to locate key information and is highly scalable. It also ensures data integrity.

A Merkle tree summarizes all the transactions in a block by producing a digital fingerprint of the entire set of transactions, thereby enabling a user to verify whether or not a transaction is included in a block.

Merkle trees are created by repeatedly hashing pairs of nodes until there is only one hash left (this hash is called the Root Hash, or the Merkle Root). They are constructed from the bottom up, from hashes of individual transactions (known as Transaction IDs).

Each leaf node is a hash of transactional data, and each non-leaf node is a hash of its previous hashes. Merkle trees are binary and therefore require an even number of leaf nodes. If the number of transactions is odd, the last hash will be duplicated once to create an even number of leaf nodes to ensure the hash can be calculated.

This is a very standard approach across blockchain projects, being used by both Bitcoin and Ethereum with great success. We therefore feel confident in adopting a similar approach with KVI.

Security Considerations

Blockchains are inherently decentralized systems which consist of different actors who act depending on their incentives and on the information that is available to them. Whenever a

new transaction gets broadcasted to the network, nodes have the option to include that transaction to their copy of their ledger or to ignore it. When the majority of the actors which comprise the network decide on a single state, consensus is achieved.
A fundamental problem in distributed computing and multi-agent systems is to achieve overall system reliability in the presence of a number of faulty processes. This often requires processes to agree on some data value that is needed during computation. These processes are described as consensus.

What happens when an actor decides to not follow the rules and to tamper with the state of his ledger?

What happens when these actors are a large part of the network, but not the majority? In order to create a secure consensus protocol, it must be fault tolerant.

Byzantine Fault Tolerance

Byzantine Fault Tolerance is the characteristic which defines a system that tolerates the class of failures that belong to the Byzantine Generals' Problem. Byzantine Failure is the most difficult class of failure modes. It implies no restrictions, and makes no assumptions about the kind of behaviour a node can have (e.g. a node can generate any kind of arbitrary data while posing as an honest actor).

Byzantine Faults are the most severe and difficult to deal with. Byzantine Fault Tolerance has been needed in airplane engine systems, nuclear power plants and pretty much any system whose actions depend on the results of a large number of sensors. Even SpaceX was considering it as a potential requirement for their systems.

Blockchains are decentralized ledgers which, by definition, are not controlled by a central authority. Due to the value stored in these ledgers, bad actors have huge economic incentives to try and cause faults. As a consequence, Byzantine Fault Tolerance is much needed.

There are a number of ways to implement Byzantine Fault Tolerance. The one we believe is best, and therefore have chosen for KVI, is:

Delegated Proof of Stake

Traditional Proof-of-Stake (PoS) was largely developed to overcome the many inherent pitfalls of the Proof-of-Work mechanism. First used by Peercoin, Proof-of-Stake places a restriction on the number of blocks that a node can validate. A node can only authenticate as many transactions as its stake in the cryptocurrency or in other words the number of coins that it holds. This solves the energy problem by limiting the computational power that one node can spend on validating a transaction.

However, PoS is prone to the "tragedy of commons," when the rewards for an individual node continue reducing, and keep dropping out of the system as a result. This makes the entire

system susceptible to a 51% attack wherein a single node or a pool of nodes could end up holding more than 51% of the total computational power of the network.

In an attempt to speed up the processing time it takes for a transaction to complete while maintaining the incentive structure for nodes, the Delegated-Proof-of-Stake (DPoS) is used by such coins as EOS, Lisk, and Steem. As a variation of the Proof-of-Stake mechanism, DPoS requires nodes to vote for other users who they trust to participate in the validation process. The nodes with the highest votes then authenticate the transactions.

This means of delegation and voting renders fairness, and avoids a 51% attack on the whole process. Furthermore, the votes are weighed in accordance with the stake that a voter node holds. This means that to avoid a loss of income and reputation, a node with a high stake will not choose someone who might be capable of malicious attacks.

Conclusion and final notes

The objective of this document was to explain the use-cases for KVI, the context within which it will be operating and the scope of the partnerships that have given rise to the need for KVI to be developed. We also aimed to explain the fundamental architectural elements to be utilised in the construction of the KVI blockchain; reassuringly, we have selected the absolute best mechanisms to ensure industry-standard builds within a highly secure and performant platform.

This Whitepaper will be further expanded and developed as we are able to share further detail about the implementation, as our Use-Case 2 develops, since this is likely to have some design impact.

If you have any further questions about the project, please contact us using the details on the cover-page.

Further Partnerships

Finally, we would like to include here some of our other partners in this venture; you can see this is already quite an extensive list; our intention is to work openly and transparently with other projects and teams in order to leverage their experience and to accelerate the KVI project towards early delivery, but also for further content delivery and integration with HAI, as discussed at the beginning of this document.

KVI Teams

Lee Ye Young
(CEO of KVI)
-Academic Degree : PhD in Global Language and Culture at Korea University
Ewha Womans University Master's Program for Performing Arts
Seoul National University Graduate School of Western Music
Career
Chairman of the Carnegie Lee foundation.
President of the Korea Music and Culture Foundation
AMP Head Professor at the Korean Institute of Education at California State University
Awards
2019 President Trump Wins Gold Medal for Charity Award
2018 Award for Music of Innovation Leaders Korea (News maker)
2016 Korea Creative Culture, Arts and Music Awards (Seoul Council Award) 2016 Korea's Best National Awards

Lee Su Min

(CMO of KVI)

-Academic Degree : Hanyang University's Department of Real Estate

Career

(Corp.) VP of asset consulting for PMG
(Corp.) Vice President of A&P Management
(Corp.) CEO of DOCM
(Corp.) CEO of Kimpo Han River City Development Project

Věnceslav Klimeš

(Senior Product Manager, Google, IT)

- Academic Degree : Masters, Czech Technical University in Prague, Computer Sciences - Algorithms : Randomized Algorithms (2012)

Career

Google : Senior Product Manager , 3 years, 4 months.
Accenture : Product Manager, 2 years 9 months
Accenture : Project manager, 2 years, 5 months. Red Hat Czech s.r.o. : Software Research, 1 years 11 months.

Alexander Vinogradow

(Principal Software Engineer, Microsoft, IT)

- Academic Degree : MS, The University of Edinburgh, Computer Sciences - Software Engineering : Formal Methods (Formal Verification) (2013)

Career

Microsoft : Principal Software Engineer , 3 years, 4 months.
Microsoft : Software Engineer, 2 years 4 months
Deloitte Technology: Penetration Tester, 2 year, 2 months.
Global logic: Application Integration Intern, 1 year.

Dima Alexeev

(BD, Metro group, Retail)

- Academic Degree : MBA, Steinbeis University Berlin Institute Corporate Responsibility Management, Internal relations: crysis management (2011)

Career

Metro group: Chief strategist, 5 years.

Volkswagen group: Strategic partners relations, 2 years 11 months.

Metro group: Internal relations manager, 3 years, 1 month.

Henkel : Business development intern, 1 year.