BuyAnyLight

YellowPaper v1.0

PUBLISHED ON SEPTEMBER 15, 2019

Enabling smart, low-cost and energy-efficient lighting solutions by harnessing the power of blockchain, AI and Big data.

AUTHORS

Johannes Eidens
Martin Heyen
Muhammad Younas
Ryan Matthew Quines
Leo Vicente
Rowell Dela Rama
Dr. Moritz Eidens

ABSTRACT

The BuyAnyLight Platform is a blockchain-based online marketplace for sourcing quality LED lighting products at competitive market rates. It uses the self-enforcement feature of smart contracts to secure the payment process and enable trustless execution of order activities within the BuyAnyLight ecosystem. Each transaction is verified through the blockchain and recorded in a decentralized ledger that cannot be altered. The smart contract is able to perform credible transactions without trusted third parties, and the transactions on the blockchain are trackable and irreversible. This yellow paper discusses the architecture and protocol for developing the infrastructure to support a decentralized LED lighting marketplace on the blockchain.

Index Terms—Blockchain, E-marketplace, smart contract, trustless

TABLE OF CONTENTS

01 INTRODUCTION

BuyAnyLight is a decentralized light sourcing platform, providing quality LED products in a seamless, secure, and cost-effective way. The BuyAnyLight Platform is built on top of Ethereum, and will provide the following options:

Light Finder
Enabling B2B and B2C users to purchase high quality LED products directly from manufacturers.

My Home
Allowing individual home-owners to get a consultation from experts and purchase the correct LED products.

My Tower
Allowing B2B users to leverage our expertise to get the right lighting solutions for their projects.

BuyAnyLight will use a mixture of decentralized nodes and centralized API servers located around the globe. Our platform will be built on the Ethereum network and will take advantage of the advanced blockchain protocol and smart contract layer. We will be using a mix of different languages, including but not limited to Solidity, C, C++, and Python to construct the BAL decentralized platform and utilize a number of “best-practices” to ensure our platform is secure and efficient.

02 MAIN ACTORS

There are three main actors in our system: sellers, experts, and buyers

Sellers

We define sellers as those manufacturing companies or retailers who are looking to attract buyers for their lighting products.

What can a seller do?

- Register on the BAL Platform
- Create and list their products
- Communicate with our experts and respond to requests for quotations
- Receive inquiries from us or buyers
- Participate in the bidding process
- Provide buyers with flexible rates depending on order size
- Generate performance reports
- Get paid in BAL tokens

BAL Experts

These are highly experienced experts who assist buyers in choosing the light products and ensuring that they get the right lighting products from the right sellers, at the right price.

What can "experts" do?

- Ask for information from the buyer
- Evaluate the information and suggest the right products and packages for the buyer
- Pass on this information to eligible sellers
- Ask a seller for a custom deal (e.g. negotiating on fees, product unit prices for buyers, etc.)
- Leave reviews on seller accounts
- Generate performance reports

Buyers

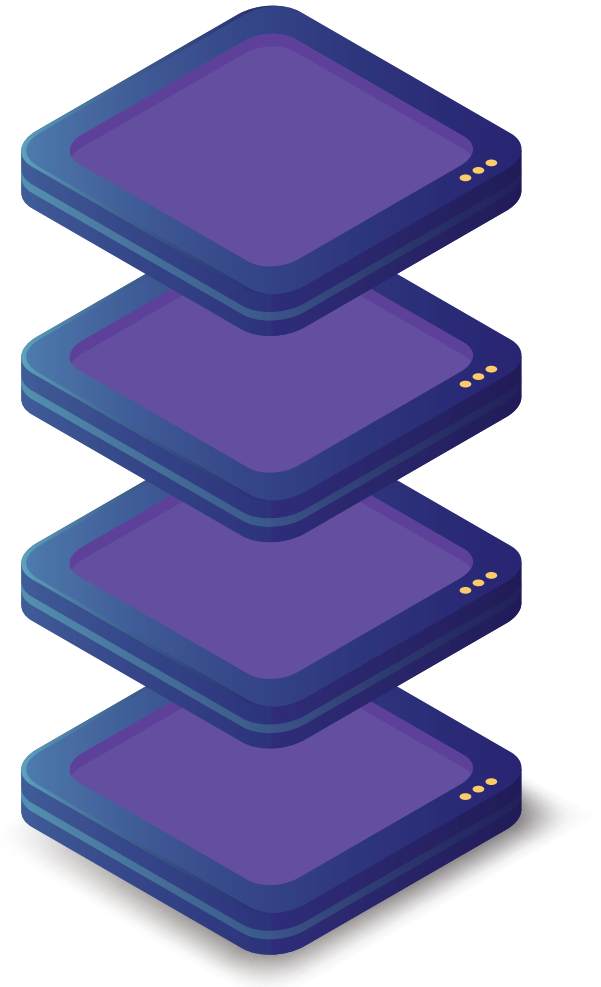
The third actor in this system is the buyer. Buyers visit the BAL Platform to buy their desired LED lighting products.

What can "buyers" do?

- Register on the BAL Platform
- Upload his/her requirements
- Communicate with the experts to select the right package
- Deposit payments
- Track and receive products

03 LAYERS

The BAL decentralized platform will have the following layers:



UI LAYER
This is the final layer that end-users will be interfacing with – this could be our web app or our mobile apps.

API LAYER
This layer connects with the blockchain layer and contains logic/code that doesn't need to be run on the smart contract layer (i.e. data storage, smart contract relaying, etc.).

SMART CONTRACT LAYER
This is the trustless layer of code that we will run on top of the blockchain layer to handle financial transactions on the blockchain.

BLOCKCHAIN LAYER
This is the core decentralized infrastructure that our platform will run on, which will be the Ethereum protocol.

04 TECHNOLOGY

BuyAnyLight Platform will be integrated to the Ethereum network, one of the largest, most secure, decentralized, and functional blockchains. Smart contracts are written in Solidity, where mathematical proofs will be used to ensure that our contracts are safe. The BAL Smart contracts will allow users to interact with our platform on-chain.

The BuyAnyLight Platform will develop smart contracts using a high-level language, as well as a few raw Solidity contracts where it makes sense. We will look to verify each contract to ensure mathematical correctness, security, and safety.

BuyAnyLight API servers will store data securely within databases that are backed up and replicated across multiple machines. All data will be stored securely and encrypted.

We are currently looking at using VueJS and Material design for our front end frameworks for the web app. BAL will also utilize either React Native or Flutter as the language for our mobile apps.

For general communication, we will be utilizing the Web3.js and Eventeum library– this provides an easy to use and direct line of communication with the core blockchain protocol.

05 INFRASTRUCTURE

BuyAnyLight will utilize AWS to run Ethereum nodes globally, using load-balancing techniques and DDoS protection to ensure our setup is robust and available all the time. We will also partner with well-known public node providers and offer our servers to the public. Our goal is to hire experts to manage and oversee the development and deployment of these nodes.

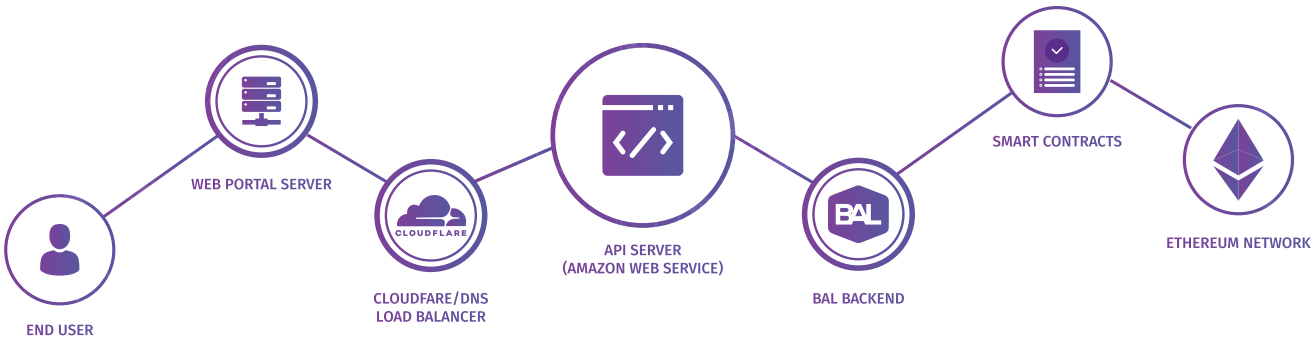
BuyAnyLight will run nodes on Gith or Parity protocol. These nodes will be secured using a 2048bit security certificate and utilize a load balancing server (with SSL between the back and front nodes). All off-chain data and logic will be run on independent AWS or Google Cloud server nodes, balanced using the AWS load-balancers and Cloudflare (for DDOS protection). This gives us a solid base infrastructure to build on that allows us to scale with our user base.

Blockchain storage

In order to minimize gas and processing costs, we are going to reduce the required blockchain storage to a minimum. The BAL token will be used for transaction signatures and metadata for encrypted storage on the blockchain and Amazon EC2 platforms. Current members, the sales portal, and object information will be stored using a regular SQL database. Storage of pictures and extended data are simply too expensive in terms of gas on the Ethereum platform at this time. We will be exploring ways to include all data on the blockchain through off-chain or scaling solutions. Furthermore, we will be using industry experts to assist with the formal verification and auditing of our code-base and infrastructure, and we have already started scouting for experts.

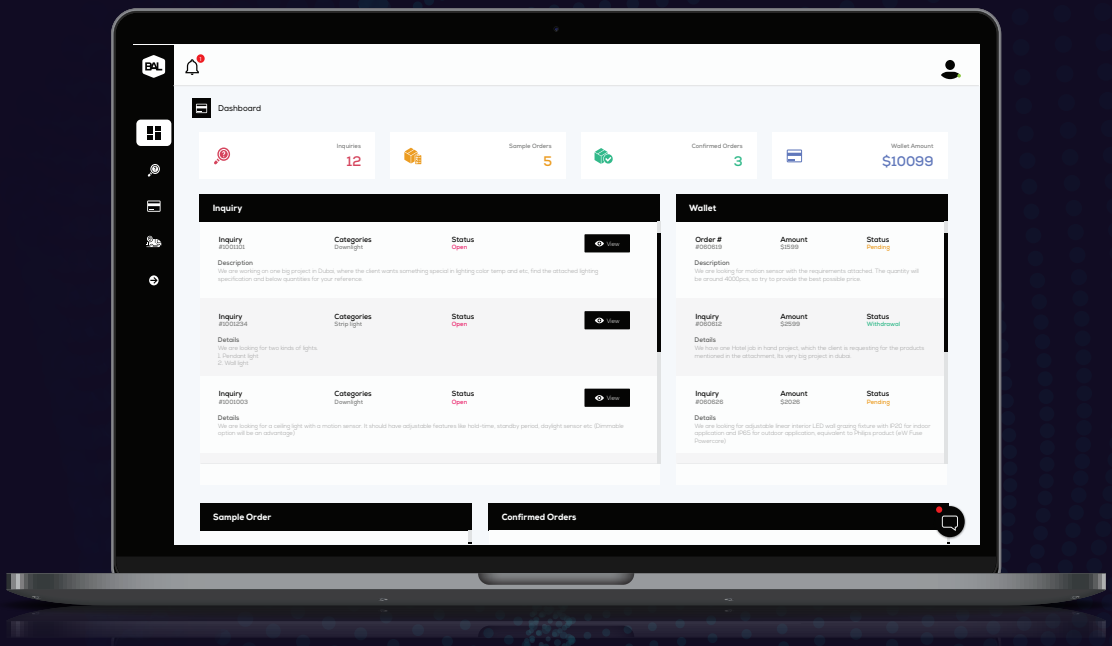
SQL DB storage

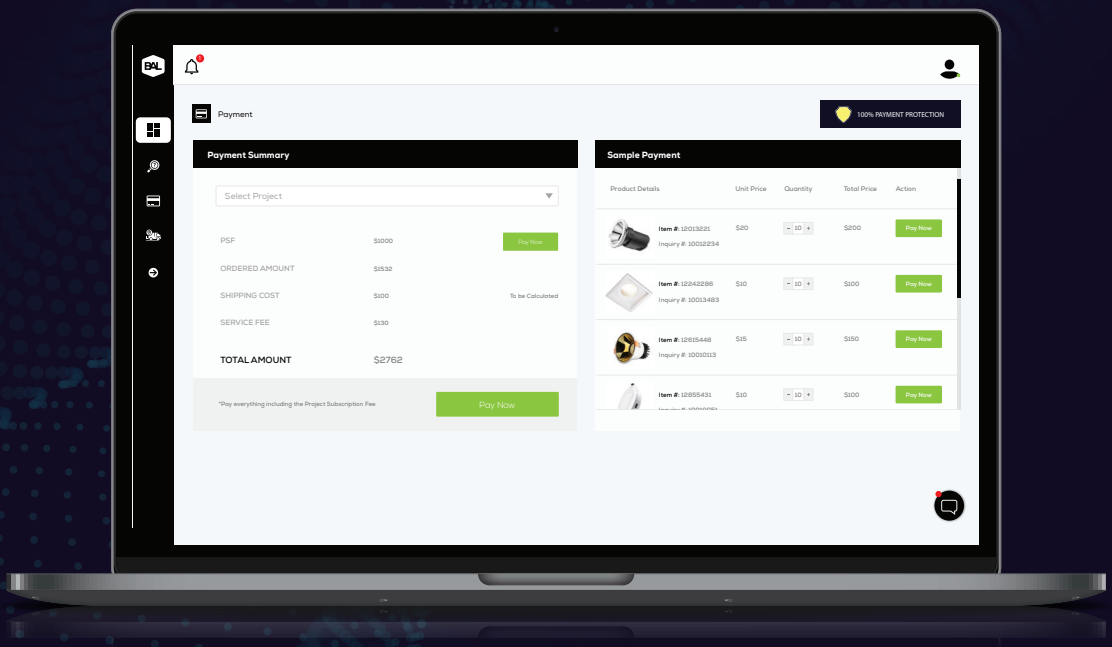
The BAL Platform will use phpMyAdmin client to manage its main SQL based database. This database will contain encrypted information about registration data for buyers, sellers, orders, transactions, and performance measures. We believe this approach will reduce the load on the blockchain side, making it more efficient than using a UI to retrieve and send data directly, which incurs additional costs.

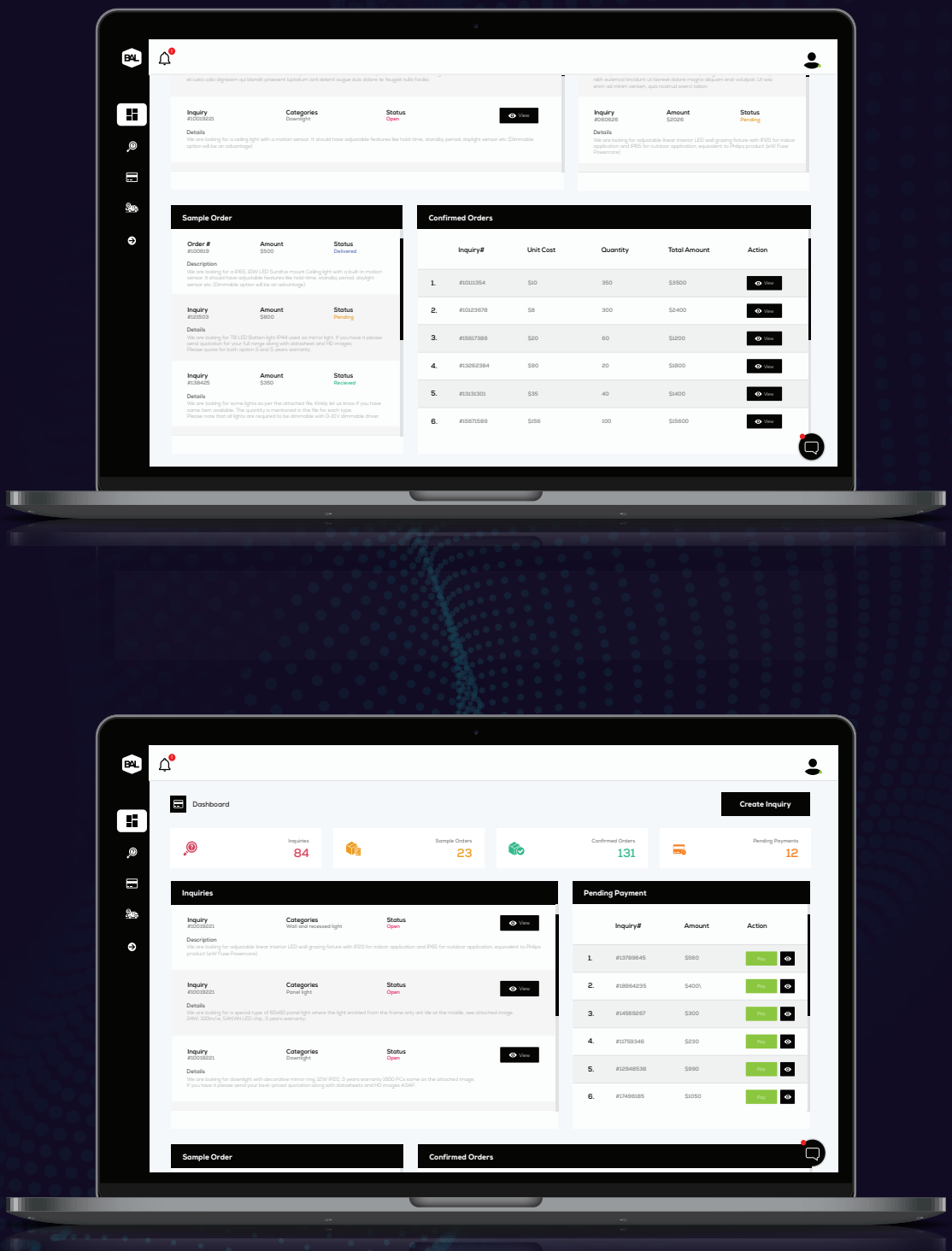


06 USER INTERFACE

For the existing platform, the preview of the user interface is below.







In future we will be employing key experts to assist with the user experience and user interface side of the decentralized BuyAnyLight Plaform. Our current platform release will use material design concepts mixed with custom designs, but we will work toward a more general set of design guidelines for our platform in the future.

Our focus is to ensure that BAL is simple to use, with appropriate functionalities for our target audience.

07 SMART CONTRACTS

BAL uses smart contracts – self-executing cryptographic code-based technology for its transactional purposes. A typical smart contract works just like a vending machine, where a buyer inserts coins, selects the item, and receives it. If the user hasn't inserted enough coins, the vending machine won't dispense the required product, and instead will insist on receiving the balance or simply cancel the transaction and return the balance. We use smart contracts to help our users perform their order activities or exchange value in a transparent, conflict freeway while avoiding the services of a middleman/third party.

By using this technology, the BAL ecosystem not only defines the rules and penalties around an agreement the same way a traditional contract does, but also automatically enforces those obligations. Furthermore, usage of smart contracts in the BAL ecosystem will ensure that in our blockchain-based distributed ledger environment, clearing and settlement will happen within seconds. In the later stages of our roadmap, we will further optimize our blockchain network so as to scale according to its growth and process, clear, and settle at different speeds depending on the market participant's needs.

BAL smart contracts will offer a way to verify and order transactions in a distributed ledger, a record of consensus that is validated and held within a network of separate nodes. Entries can be altered but not deleted from the blockchain, which is a distributed ledger-based on the Ethereum platform. All transactions will have an auditable trail and a traceable digital fingerprint. The data on the ledger is pervasive and persistent, creating a reliable "transaction cloud" where data cannot be lost.

BuyAnyLight uses smart contract technologies that are deterministic exchange mechanisms controlled by digital means to carry out the direct transaction of value between untrusted agents. BAL will use this technology to facilitate, verify, and enforce the negotiation or performance of economically laden procedural instructions and circumvent censorship, collusion, and counterparty risk.

BuyAnyLight will utilize three core smart contracts, with the ability to add additional contracts in the future. These contracts are:

1. **Core Escrow Contract** – this handles the order activity by serving as an escrow logic and allows for signed requests to make repayments on a transaction once the order has correctly been processed and both parties have agreed on the particulars to their satisfaction.
2. **Token contract** – this will be a standard ERC20-like token contract for the BAL token on top of the Ethereum blockchain network.
3. **Exchange Contract** – this contract will allow the instant liquidation of collateral at a certain price and will act as an "order book" to cover liquidations.

Core Escrow Contract

This smart contract will handle the day-to-day and marketplace related activities of the BuyAnyLight Platform on the blockchain. It will further create an opportunity for buyers and sellers to optimize settlement for participants and perform order activities in a decentralized and trustless manner. The Core Escrow Contract will perform the following tasks:

1 **Creation and funding of order activity** – buyers will interact directly with our smart contract, sending their crypto BAL tokens and creating an on-chain order. The creation process must be signed by the core BAL key to ensure that the order is processed and confirmed from the BAL Platform.

2 **Cancelling an order** – an order can be cancelled if the status remains "Pending" and seven days have passed since the creation of the order.

3 **Accepting an order** – when an order request becomes funded, the BAL Platform will assign it to a specific re tailer or manufacturer based on the requirements by sending a message to the contract, which accepts the order. This makes the order contract live, once the seller has accepted it.

4 **Clearing and processing** – once the order has been processed and the seller has delivered the products as per the buyer's requirements, and the order has been marked as cleared by the BAL quality assurance team, the BAL Platform will mark this against the amount in the contract to update the on-chain tracking and processing of the payment to the seller.

5 **Liquidation and release** - BAL will request the smart contract to liquidate and release the order amount to the seller. This is all done internally and is triggered by our oracle.

All liquidated BAL tokens are sent to the Exchange contract, which is where we convert the crypto into a fiat to repay the seller.

Token Contract

The BuyAnyLight Platform will generate a token smart contract for creation of its ERC20 BAL tokens. Tokens will be allocated to donors of Ether through a devised, predetermined structure. There will be two planned purchasing stages: a pre-sale and a crowd sale. The limited pre-sale will be used to generate initial funds for crowd sale preparation and development. The crowd sale will allow for maximum user adoption and token distribution.

In the long run, the BAL Platform will operate its own native dedicated encrypted wallet. It will also give users the ease of mind to utilize existing wallet platforms and blockchain explorer APIs to maintain a current balance of user tokens.

As this smart contract system evolves, the infrastructure will need to be modified to keep pace with emergent upgrades to the blockchain protocol. From the standpoint of data storage, our platform is blockchain neutral, and the calls it makes to write to the platform are not married to any given coin ecosystem.

The BAL API is Platform-agnostic, and the data can be stored on any blockchain or server. The key here is that the proof of concept is complete, and BAL engineering will focus on keeping costs low and throughput high for users during the benchmarking phase of the deployment.

Token Contract Source Code

```
pragma solidity 0.5.11;

/**
 * @title SafeMath
 * @dev Math operations with safety checks that throw on error
 */
library SafeMath {
    function mul(uint256 a, uint256 b) internal pure returns (uint256) {
        uint256 c = a * b;
        assert(a == 0 || c / a == b);
        return c;
    }

    function div(uint256 a, uint256 b) internal pure returns (uint256) {
        // assert(b > 0); // Solidity automatically throws when dividing by 0
        uint256 c = a / b;
        // assert(a == b * c + a % b); // There is no case in which this doesn't
hold
        return c;
    }

    function sub(uint256 a, uint256 b) internal pure returns (uint256) {
        assert(b <= a);
        return a - b;
    }

    function add(uint256 a, uint256 b) internal pure returns (uint256) {
        uint256 c = a + b;
        assert(c >= a);
        return c;
    }
}


/**
 * @title Ownable
 * @dev The Ownable contract has an owner address, and provides basic authorization
control
 * functions, this simplifies the implementation of "user permissions".
 */
contract Ownable {

    address public owner;

    event OwnershipTransferred(address indexed previousOwner, address indexed ne-
wOwner);

    /**
     * @dev The Ownable constructor sets the original `owner` of the contract to
the sender
     * account.
     */
```

```
    constructor() public {
        owner = msg.sender;
    }


    /**
     * @dev Throws if called by any account other than the owner.
     */
    modifier onlyOwner() {
        require(msg.sender == owner);
        _;
    }

    /**
     * @dev Allows the current owner to transfer control of the contract to a ne-
wOwner.
     * @param newOwner The address to transfer ownership to.
     */
    function transferOwnership(address newOwner) onlyOwner public {
        require(newOwner != address(0));
        emit OwnershipTransferred(owner, newOwner);
        owner = newOwner;
    }
}

/**
 * @title ERC20Basic
 * @dev Simpler version of ERC20 interface
 * @dev see https://github.com/ethereum/EIPs/issues/179
 */
contract ERC20Basic {
    uint256 public totalSupply;
    function balanceOf(address who) public view returns (uint256);
    function transfer(address to, uint256 value) public returns (bool);
    event Transfer(address indexed from, address indexed to, uint256 value);
}


/**
 * @title Basic token
 * @dev Basic version of StandardToken, with no allowances.
 */
contract BasicToken is ERC20Basic {

    using SafeMath for uint256;

    mapping(address => uint256) internal balances;

   /**
     * @dev transfer token for a specified address
     * @param _to The address to transfer to.
     * @param _value The amount to be transferred.
     */
   function transfer(address _to, uint256 _value) public returns (bool) {
        require(_to != address(0) && _to != address(this));
```

```
        // SafeMath.sub will throw if there is not enough balance.
        balances[msg.sender] = balances[msg.sender].sub(_value);
        balances[_to] = balances[_to].add(_value);
        emit Transfer(msg.sender, _to, _value);
        return true;
    }

    /**
     * @dev Gets the balance of the specified address.
     * @param _owner The address to query the the balance of.
     * @return An uint256 representing the amount owned by the passed address.
     */
    function balanceOf(address _owner) public view returns (uint256 balance) {
        return balances[_owner];
    }
}

/**
 * @title ERC20 interface
 * @dev see https://github.com/ethereum/EIPs/issues/20
 */
contract ERC20 is ERC20Basic {
    function allowance(address owner, address spender) public view returns
(uint256);
    function transferFrom(address from, address to, uint256 value) public returns
(bool);
    function approve(address spender, uint256 value) public returns (bool);
    event Approval(address indexed owner, address indexed spender, uint256 value);
}


/**
 * @title Standard ERC20 token
 *
 * @dev Implementation of the basic standard token.
 * @dev https://github.com/ethereum/EIPs/issues/20
 * @dev Based on code by FirstBlood: https://github.com/Firstbloodio/token/blob/
master/smart_contract/FirstBloodToken.sol
 */
contract StandardToken is ERC20, BasicToken {

    mapping (address => mapping (address => uint256)) internal allowed;

    /**
     * @dev Transfer tokens from one address to another
     * @param _from address The address which you want to send tokens from
     * @param _to address The address which you want to transfer to
     * @param _value uint256 the amount of tokens to be transferred
     */
    function transferFrom(address _from, address _to, uint256 _value) public
    returns (bool) {
        require(_to != address(0) && _to != address(this));

       uint256 _allowance = allowed[_from][msg.sender];

        // Check is not needed because sub(_allowance, _value) will already throw
if this condition is not met
        // require (_value <= _allowance);
```

```
        balances[_from] = balances[_from].sub(_value);
        balances[_to] = balances[_to].add(_value);
        allowed[_from][msg.sender] = _allowance.sub(_value);
        emit Transfer(_from, _to, _value);
        return true;
    }

   /**
    * @dev Approve the passed address to spend the specified amount of tokens on be-
half of msg.sender.
    *
    * Beware that changing an allowance with this method brings the risk that some-
one may use both the old
    * and the new allowance by unfortunate transaction ordering. One possible solu-
tion to mitigate this
    * race condition is to first reduce the spender's allowance to 0 and set the de-
sired value afterwards:
    * https://github.com/ethereum/EIPs/issues/20#issuecomment-263524729
    * @param _spender The address which will spend the funds.
    * @param _value The amount of tokens to be spent.
    */
    function approve(address _spender, uint256 _value) public returns (bool) {
        allowed[msg.sender][_spender] = _value;
        emit Approval(msg.sender, _spender, _value);
        return true;
    }

    /**
     * @dev Function to check the amount of tokens that an owner allowed to a
spender.
     * @param _owner address The address which owns the funds.
     * @param _spender address The address which will spend the funds.
     * @return A uint256 specifying the amount of tokens still available for the
spender.
     */
    function allowance(address _owner, address _spender) public view returns
(uint256 remaining) {
        return allowed[_owner][_spender];
    }

    /**
     * approve should be called when allowed[_spender] == 0. To increment
     * allowed value is better to use this function to avoid 2 calls (and wait un-
til
     * the first transaction is mined)
     * From MonolithDAO Token.sol
     */
    function increaseApproval (address _spender, uint _addedValue) public returns
(bool success) {
        allowed[msg.sender][_spender] = allowed[msg.sender][_spender].add(_added-
Value);
        emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
        return true;
    }
```

```
    function decreaseApproval (address _spender, uint _subtractedValue) public re-
turns (bool success) {
        uint oldValue = allowed[msg.sender][_spender];
        if (_subtractedValue > oldValue) {
            allowed[msg.sender][_spender] = 0;
        } else {
            allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
        }
        emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
        return true;
    }
}


/**
 * @title Burnable Token
 * @dev Token that can be irreversibly burned (destroyed).
 */
contract BurnableToken is StandardToken {

    event Burn(address indexed burner, uint256 value);

    /**
     * @dev Burns a specific amount of tokens.
     * @param _value The amount of token to be burned.
     */
    function burn(uint256 _value) public {
        require(_value > 0);
        require(_value <= balances[msg.sender]);
        // no need to require value <= totalSupply, since that would imply the
        // sender's balance is greater than the totalSupply, which *should* be an
assertion failure

        address burner = msg.sender;
        balances[burner] = balances[burner].sub(_value);
        totalSupply = totalSupply.sub(_value);
        emit Burn(burner, _value);
        emit Transfer(burner, address(0), _value);
    }
}

interface tokenRecipient {
    function receiveApproval(address _from, uint256 _value, bytes calldata _extra-
Data) external;
}

contract BALToken is BurnableToken, Ownable {

    string public constant name = "BAL Token";
    string public constant symbol = "BAL";
    uint public constant decimals = 18;
    // there is no problem in using * here instead of .mul()
    uint256 public constant initialSupply = 55000000 * (10 ** uint256(decimals));
```

```
    // Constructors
    constructor () public {
        totalSupply = initialSupply;
        balances[msg.sender] = initialSupply; // Send all tokens to owner
        emit Transfer(address(0), msg.sender, initialSupply);
    }

    function approveAndCall(address _spender, uint256 _value, bytes calldata _ex-
traData)
        external
        returns (bool success)
    {
        tokenRecipient spender = tokenRecipient(_spender);
        if (approve(_spender, _value)) {
            spender.receiveApproval(msg.sender, _value, _extraData);
            return true;
        }
    }

    function transferAnyERC20Token(address _tokenAddress, address _to, uint _
amount) public onlyOwner {
        ERC20(_tokenAddress).transfer(_to, _amount);
    }
}
```

BAL Token
This repository contains the BAL (BuyAnyLight) Token deployed on the Ethereum network.

View on Etherscan
Audited by Blockhain Consilium

About the Token
The token is compliant with ERC20 standard, with a few more improvements:

Recovery feature for unsold tokens
One step transfer to contract to save gas and time
Burnable functionality

Zeppelin Code
Contract is based on OpenZeppelin's `SafeMath`, `Ownable`, basic version of `StandardToken` and `BurnableToken` logic.

08 FIAT PARTNER

The BAL Platform uses a fiat-partner, i.e, www.foloosi.com, which works as payment gateway enabling our users to deposit funds into their BAL account for the following purposes:

1. To fund an offer toward a respective order activity.
2. To enable the purchase of BAL tokens directly from fiat money.
3. To enable sellers to receive and exchange BAL tokens and fiat money seamlessly.

In the current marketplace, we are already accepting payment via credit card and QR Codes.

We will look at eventually tokenizing fiat deposits and withdrawals to allow the tracking of these transactions on the blockchain. This provides a more transparent and auditable trail; however, it is not a type of “stable coin”.

09 WORKING OF THE LEDGER

The commercial activity ledger is the backbone of the BuyAnyLight ecosystem and provides the decentralized capability to validate customer and vendor activity. The activity ledger facilitates trust in all order activities. Any party who can attest to activities by an individual user or business can contribute validation data to the ledger, and any party who can access the ledger entries can potentially get value from the data, in accordance with the data-sharing terms specified by the smart contract based agreement between participants.

BuyAnyLight transactions will be done on the blockchain. Each block added onto the blockchain from BuyAnyLight will contain anonymized information about an organization, product, or individual. The data encryption is done in a way that allows only specific groups of authorized parties to pull data relevant for their own use. For all other parties the data will be untraceable to specific individuals but still valuable for statistical and data aggregation purposes. This gives everyone control over what data can be created, shared, and accessed, all using the Ethereum blockchain and smart contracts.

10 WORKING OF THE PLATFORM

The BAL Platform will pair sellers and buyers through the blockchain – initially we will accept fiat currencies as well. We will accept USD, BTC, ETH and BAL (the BAL token).

Buyers will be required to register and select from the various protocols, i.e., My Home, My Tower and LightFinder, and get consulting services such as General Lighting Consultation, Lux Calculation, and 3D design, all based on their particular set of requirements. After they have registered and submitted their application via our user-friendly interface, the role of the BAL Platform is to find them the right LED products at competitive market rates. The platform uses a combination of AI-based algorithms, including but not limited to BFS, DFS, DLS, UCS, and A* algorithm.

The BAL Platform initially will not operate on its own native wallet, but will use existing wallet platforms and blockchain explorer APIs to maintain a current balance of user tokens. Later on, however, it will create a mobile and desktop version of the wallet.

Furthermore, the BAL API is platform-agnostic, and the data can be stored on any blockchain or server; the key is that the proof of concept is complete. BuyAnyLight Platform engineering is focusing on keeping costs low and throughput high for users during the benchmarking phase of the deployment.

11 DEPLOYMENT PLAN

Combining all the above aspects, BuyAnyLight acts as a platform for serving customer-oriented needs in flexible transaction formats (i.e., buy, sell, receive). As the platform provider, BuyAnyLight's primary mission is to enhance the existing infrastructure by further deploying the future core framework stage by stage, while allowing developers to create add-on features and services to their existing online platform. The BuyAnyLight Platform has no known predecessor in the world of cryptocurrency platforms. As such, the BuyAnyLight core code serves Business-to-Customer, Customer-to-Customer, and Business-to-Business interactions with a set of simple APIs that are already available for review. BuyAnyLight realizes that there is value in online marketplaces such as eBay and Amazon, which have revolutionized the ecommerce marketplace, and it intends to offer the same solutions specifically for LED lighting products. Although BuyAnyLight does not follow the same business model as these platforms, it uses elements of them. BuyAnyLight plans to create a decentralized lighting based economy that allows buyers and sellers to receive quality lighting products at not only a reduced cost, but also more efficiently and with less stress.

In light of the innovation and open market niche involved in launching this platform, BuyAnyLight benefits from a first-mover advantage -- it is the continuation of an existing company, Almani Lighting LLC, which is a leading lighting company based in UAE. Almani Lighting LLC has a successful track record of industry expertise and quality lighting products in the European and MENA regions. The need for BAL was identified from the business model (Almani Lighting LLC) that shows, for the lighting industry (specifically the procurement and sourcing process), there is a need of a blockchain based sourcing platform i.e. BuyAnyLight. So, this is not your traditional crowd sale starting from scratch — it is not a startup that can only provide speculative value for its tokens, since you can see and use the existing platform for yourself.

Initial Development Efforts

The objectives for BuyAnyLight have been refined not from speculative ideas about how a program should work, but they are the result of experience and industry observation, plus the leadership of experienced founders. This business model and the current software are the result of several iterations of the platform over the years. BuyAnyLight already has a working platform that will integrate design and protocol updates, particularly those within blockchain technology. It supports hundreds of manufacturers and retailers from all over the world and has a successful track record of providing quality LED solutions to a huge client base in the Middle East and Germany.

Growth Plan

As part of the process of building a decentralized ecosystem on a global level, BuyAnyLight has tapped its established user groups from Germany and UAE in particular, since the platform has the infrastructure and regional headquarters in these locations. Many of the BuyAnyLight promoters in these areas are already familiar with the platform.

BuyAnyLight will further identify key light manufacturers and retailers that have a significant global footprint, so that the platform can scale seamlessly and significant amounts of lighting products can be tokenized into BuyAnyLight. In addition, the marketing team will identify the participants, i.e., buyers and sellers who will participate in the token sale process. Moreover, BuyAnyLight is already in discussions with key representatives who are awaiting our final roll-out, which will give it immediate participation on a global level. BuyAnyLight works extensively with lighting manufacturers, and will continue to provide updates on which ones are integrated with the platform. This will be done at the local, regional, national, and global level.

12 GLOSSARY

Artificial Intelligence (AI) - This is intelligence demonstrated by machines. It is the opposite of natural intelligence (NI), which is intelligence demonstrated by humans or animals.

API - A set of programming libraries and functions that allow outsiders to interact with a given system infrastructure.

BFS - Breadth-first search, an AI algorithm.

BAL - BuyAnyLight.

BAL Token - The native token of the BAL Platform that is deployed on the Ethereum network.

Blockchain - A blockchain is a continuously growing list of records, called blocks, which are linked and secured using cryptography.

DFS - Depth-first search, a type of algorithm used in AI.

DLS - Depth limited search.

ERC-20 token - A token that is created on the Ethereum platform via Ethereum token smart contract, which easily allows point-to-point token exchange.

KYC - Know Your Customer, a set of procedures to determine with a high degree of certainty the identity of a participant.

LED - A light-emitting diode (LED) is a semiconductor light source that emits light when a current flows through it.

Smart Contract - An automatically enforced agreement among two or more parties in the ecosystem, mapping a set of activities that are to be executed to ledger operations.

Token Exchange - A community established exchange rate, assigning value to a given activity or service.

UCS - Uniform cost search.

Wallet - A cryptocurrency wallet is a software program that stores private and public keys and interacts with various blockchains to enable users to send and receive digital currency and monitor their balance.

13 DISCLAIMER RISKS AND CHANGES TO THE PLATFORM

At the date of the token sale, the platform has specific functionality and utility. However, the platform may undergo significant changes. Various features of the platform, including its terms and conditions, fees, structure, purpose, consensus protocol, algorithm, source codes, infrastructure design and other technical specifications and parameters, may be updated and changed frequently without notice.

Further, the platform may encounter difficulties during development, including financial, resourcing and technical difficulties. These difficulties are unpredictable and may be unresolvable. Development of the platform may therefore fail, terminate or be delayed at any time for any reason. Particular features of the platform may never be realized. Such failure, termination or delay is likely to reduce and may completely obliterate any existing utility that tokens hold.

Incomplete information regarding the platform
You will not have full access to all the information relevant to the company and/or the platform. The company is not required to update you on the progress of the platform.

Lack of operating history of the company
The company is a newly formed entity, and has no operating history or track record that could be used (on its own) to evaluate its ability to deliver the platform.

No governance rights attaching to tokens
Tokens confer no governance rights of any kind with respect to the platform or the company. Accordingly, subject to other written arrangements to the contrary, all decisions involving the platform will be made by the company, including decisions to discontinue the platform. These decisions could adversely affect the platform and the utility of any tokens that you hold.

No assets or property underlying tokens
Tokens confer no ownership interest in any assets or property. Nothing underpins the value of tokens other than the prospect of platform access.

Replication, modification or enhancement arising from open-source nature of platform
The platform may have various portions that will be open source in nature. This could allow anyone to replicate, modify or enhance the technologies that underlie the platform, and readily create competitors for the platform. These competitors may be more successful than the platform.

Potential for misuse of the platform
Services which may be banned, restricted or deemed immoral in certain jurisdictions may potentially use the platform. As a result, governmental authorities or regulators may take action against the platform. This may deter users from using the platform, or may involve the complete shutdown of the platform.

Reliance on third-party contractors
Development of tokens and the platform, and the operation of the token sale, will require third-party contractors with particular expertise in Ethereum and blockchain technology. The availability of such contractors is limited. There may not be sufficient (or any) such contractors available on terms deemed acceptable by the company. The costs associated with any such contractors may be significantly greater than currently estimated. Further, the quality, reliability and timely delivery of services by such contractors may vary significantly.

Utility of tokens depends on the platform
The utility of tokens depends on the success of the platform, if developed. The platform may not be popular or widely used after launch. In the long term, the platform may fail to attract a critical mass of users. The platform may be merged with other projects. Various circumstances, including technical advancement and competitors, may render the platform obsolete.

Speculation may drive demand for tokens

Tokens are transferable in accordance with these terms and conditions. Accordingly, demand for tokens may be partially or wholly driven by speculation. Speculation may continue to drive demand for tokens even after the launch of the platform.

Volatility of tokens

The circulation of tokens is not the responsibility of the company, and the company does not presently intend to support or otherwise facilitate the secondary trading of tokens. As a result, tokens may not circulate freely or widely, and may not be listed on any secondary markets. Even if tokens do circulate on secondary markets, the value of tokens may be highly volatile. Factors such as perceptions of the company, delays in the development of the platform, fluctuations in comparable projects and token sales, market dynamics, regulatory actions and changes, technical advancements, as well as broader economic and political factors, may cause the value of tokens to change significantly over a short period of time.

In addition, there may be insufficient liquidity to support an active market in tokens, or the market in tokens may become susceptible to market manipulation.

Volatility of ETH, other digital assets and fiat currencies

Payments are made in ETH. The company may hold the proceeds in ETH, other digital assets or fiat currencies, or a combination of any of them. The value of these fiat currencies and digital assets may fluctuate significantly over a short period of time as a result of market dynamics, regulatory actions and changes, technical advancements, exchange availability, and broader economic and political factors. This volatility is likely to impact the funding that is available for developing the platform and may affect the utility of tokens.

Concentration of token ownership

At any point in time, one or more persons may directly or indirectly control significant portions of the total supply of tokens. Acting individually or in concert, these holders may have significant influence over the platform. They may make decisions that are not in your best interest as a holder of tokens.

Legal status of token, token sale, and platform is pending

The laws of various jurisdictions may apply to tokens, the token sale and the platform. The application of these laws and regulations to tokens, the token sale and the platform is, in many cases, largely untested, and is subject to change without notice. In particular, any current governmental or regulatory tolerance of cryptographic tokens or cryptocurrencies can change rapidly, and tokens may at any time be deemed to be a security, investment, asset or money by governmental authorities or regulators.

At this stage, the company expects that the platform will be regulated in a number of key markets. This will require licensing that cannot be assured. This means that the platform may not be available in certain markets, or at all. This could require fundamental restructuring of the platform. In a worst case, this could render the tokens worthless because the platform cannot be executed.

The company may receive formal or informal queries, notices, requests or warnings by governmental authorities and regulators. Action may be taken by governmental authorities and regulators against the company or the platform. As a result of such events, the company may be required to discontinue the token sale and/or the platform. You may also be subject to governmental or regulatory action by participating in the token sale, holding tokens and/or using the platform.

Tax treatment and accounting

The company, purchaser and transactions in relation to the tokens and/or the platform may be subject to the tax laws and regulations in any applicable jurisdictions. The tax treatment and accounting of transactions in relation to the tokens and/or the platform are uncertain and a largely untested area of law and practice that is subject to prospective and retroactive changes without notice. Tax treatment of cryptographic tokens and cryptocurrencies may vary amongst jurisdictions.

Your participation in the token sale or use of the platform as a result of or in connection with any purchase, grant, delivery, exercise, vesting, distribution, activation, holding, use, appreciation, conversion, sale, exchange, redemption, assignment, transfer, disposal, may attract taxes either now or in the future.

The company may receive formal or informal queries, notices, requests, or summons from tax authorities and as a result the company may be required to furnish certain information about the token sale and/or the platform. You must seek independent professional advice on the tax implications in relation to the token sale, use of the platform and/or any other transactions for your particular situation.

Reliance on the internet
Tokens, the token sale, and the platform rely heavily on the internet. However, the public nature of the internet means that either parts of the internet or the entire internet may be unreliable or unavailable at any given time. Further, interruption, delay, corruption or loss of data, or the loss of confidentiality in the transmission of data, may occur when transmitting data via the internet.

Reliance on Ethereum
Tokens, the token sale, and the platform rely on Ethereum. Ethereum is open source software that is built upon experimental technology, namely blockchain. Risks arising from this reliance include (but are not limited to):

(a) the existence of technical flaws in Ethereum;
(b) targeting of Ethereum by malicious persons;
(c) changes in Ethereum's consensus protocol or algorithms;
(d) decreased community or miner support for Ethereum;
(e) rapid fluctuations in the value of ETH;
(f) the existence or development of competing networks and platforms;
(g) the existence or development of Forked versions of Ethereum;
(h) flaws in the Solidity scripting language;
(i) disputes between Ethereum developers, miners and/or users; and
(j) regulatory action against Ethereum developers, miners and/or users.

Cryptographic advancements and developments in cryptographic technologies and techniques, including the advancement of artificial intelligence and/or quantum computing, pose security risks to all cryptography-based systems including tokens and the platform. Applying these technologies and techniques to tokens and/or the platform may result in theft, loss, disappearance, destruction, devaluation or other compromises of tokens, the platform or your data.

Source code changes and flaws
The various source codes used in the token sale and (if and when developed) the platform are subject to change and may at any time contain one or more defects, weaknesses, inconsistencies, errors or bugs.

No anonymity when using Ethereum
Your participation in the token sale, holding and transfer of tokens, and/or use of the platform (if and when developed) will not be anonymous. Your address and such participation will be recorded on an unpermissioned blockchain, namely Ethereum. It is possible to match addresses to identities.

Inadequate computing resources
The token sale and the platform will require intensive computing resources. The demand for these resources may exceed the company's estimates. Ultimately, the company's resources may prove inadequate to support the token sale and/or develop the platform, which may affect the delivery and/or utility of tokens.

Loss of private key is permanent and irreversible
You alone are responsible for securing your private key. Losing control of your private key will permanently and irreversibly deny you access to your tokens. Neither the company nor any other person will be able to retrieve or protect your tokens. Once lost, you will not be able to transfer your tokens to any other address or wallet. You will not be able to realize any value or utility that the token may hold now or in the future.

Targeting of tokens, the token sale, the platform, and the company by malicious persons. Tokens, the token sale, the platform (if and when developed) and the company may be targeted by malicious persons who may attempt to steal tokens or the proceeds, or otherwise intervene in the token sale, the platform, or the company. This includes (but is not limited to) interventions by way of:

(a) distributed denial of service;
(b) sybil attacks;
(c) phishing;
(d) social engineering;
(e) hacking;
(f) smurfing;
(g) malware;
(h) double spending;
(i) majority-mining, consensus-based, or other mining attacks;
(j) misinformation campaigns; and
(k) spoofing.

Tokens, the token sale and the platform may also be vulnerable to exploitation of vulnerabilities in smart contracts and other code, as well as to human error. This could result in significant loss and/or other impacts that may materially affect your interests.

Targeting of purchaser by malicious persons. Malicious entities may target you in an attempt to steal any tokens or cryptocurrencies that you may hold, or to claim any tokens that you may have purchased. This may involve unauthorized access to your Digital Wallet, your private keys, your cryptocurrency addresses, your email or social media accounts, as well as unauthorized access to your computer, smartphone, and any other devices that you may use. You alone are responsible for protecting yourself against such actions.

Jurisdiction related risks

Residents, tax residents, or persons having a relevant connection with certain jurisdictions are excluded from the token sale. Changes in your place of domicile or the applicable law may result in you violating any legal or regulatory requirements of your applicable jurisdiction. You are responsible for ensuring that the delivery, holding, use or exchange of tokens is and remains lawful despite changes to applicable laws, your residence, and circumstances.

Limitations on Liability

To the fullest extent permitted by applicable law: (i) in no event will the company be liable for any indirect, special, incidental, consequential, or exemplary damages of any kind (including, but not limited to, where related to loss of revenue, income or profits, loss of use or data, or damages for business interruption) arising out of or in any way related to the purchase of BAL tokens or otherwise related to these terms, regardless of the form of action, whether based in contract, tort (including, but not limited to, simple negligence, whether active, passive or imputed), or any other legal or equitable theory (even if the party has been advised of the possibility of such damages and regardless of whether such damages were foreseeable); and (ii) in no event will the aggregate liability of the company, whether in contract, warranty, tort (including negligence, whether active, passive or imputed), or other theory, arising out of or relating to these terms exceed the amount purchaser pays to the company for the BAL tokens.

Binding Arbitration

Except for any disputes, claims, suits, actions, causes of action, demands or proceedings (collectively, "Disputes") in which either party seeks injunctive or other equitable relief for the alleged unlawful use of intellectual property, including, without limitation, copyrights, trademarks, trade names, logos, trade secrets or patents, the purchaser and the company (i) waive the purchaser's and the company's respective rights to have any and all disputes arising from or related to the terms of the purchase of BAL tokens resolved in a court, and (ii) waive the purchaser's and the company's respective rights to a jury trial. Instead, the purchaser and the company will arbitrate disputes through binding arbitration (which is the referral of a dispute to one or more persons charged with reviewing the dispute and making a final and binding determination to resolve it instead of having the dispute decided by a judge or jury in court).

No Class Arbitrations, Class Actions, or Representative Actions

Any dispute arising out of or related to the purchase of BAL tokens is personal to the purchaser and the company and will be resolved solely through individual arbitration and will not be brought as a class arbitration, class action or any other type of representative proceeding. There will be no class arbitration or arbitration in which an individual attempts to resolve a dispute as a representative of another individual or group of individuals. Further, a dispute cannot be brought as a class or other type of representative action, whether within or outside of arbitration, or on behalf of any other individual or group of individuals.

CONTACT US

REGISTER NOW!

BuyAnyLight.com
Dubai Investment Park 1,
Opp. Green Community East,
Showroom 10, P.O. BOX 3219
Dubai, United Arab Emirates

+971 4 887 3265

ieo@buyanylight.com
info@buyanylight.com
invest@buyanylight.com
Call or email and let's get talking!

www.buyanylight.com