SOFTWARE DEVELOPMENT AGREEMENT

This Software Development Agreement ("Agreement") is made and effective this _July 16, 2021_, by and between High Throughput Biology Inc. ("Buyer") and _Victor Ciobanu_ ("Developer"), who has a residential address at _Chisinau, Moldova_.

NOW THEREFORE, in consideration of the mutual promises herein contained, the parties hereto agree as follows:

1. Duties and Responsibilities.

Developer shall serve as a contractor of Buyer and shall design, develop, and implement applications software (the "Software") according to the functional specifications and related information, if any, attached hereto as Exhibit A and incorporated herein by this reference (the "Specifications") and as more fully set forth in this Agreement. Developer acknowledges that it has been contracted for this specific task, and that it shall report all findings and make all recommendations directly to the management of Buyer. The Software, including all versions in either source code or object code form, shall be delivered to Buyer not later than _September 16, 2021_ [Completion Date].

2. Ownership of Software.

Developer agrees that the development of the Software is "work made for hire" within the meaning of the Copyright Act of 1976, as amended, and that the Software shall be the sole property of Buyer. Developer hereby assigns to Buyer, without further compensation, all of its right, title and interest in and to the Software and any and all related patents, patent applications, copyrights, copyright applications, trademarks and trade names in the United States and elsewhere. Developer will keep and maintain adequate and current written records with respect to the Software (in the form of notes, sketches, drawings and as may otherwise be specified by Buyer), which records shall be available to and remain the sole property of Buyer at all times. All versions of the Software shall contain Buyer's conspicuous notice of copyright. Developer will assist Buyer in obtaining and enforcing patent, copyright and other forms of legal protection for the Software in any country. Upon request, Developer will sign all applications, assignments, instruments and papers and perform all acts necessary or desired by Buyer to assign the Software fully and completely to Buyer and to enable Buyer, its successors, assigns and nominees, to secure and enjoy the full and exclusive benefits and advantages thereof.

3. Compensation.

A. Buyer shall pay Developer _$2000_ [Purchase Price] as follows: _three milestone payments of $1100, $400, and $500 through PayPal_ [Payment Method].

B. No additional compensation will be provided by Buyer to Developer for this project.

4. Independent Contractor.

Developer is acting as an independent contractor with respect to the services provided to Buyer. Neither Developer nor the employees of the Developer performing services for Buyer will be considered employees or agents of Buyer. Buyer will not be responsible for Developer's acts or the acts of Developer's employees while performing services under this Agreement. Nothing contained in this Agreement shall be construed to imply a joint venture, partnership or principal-agent relationship between the parties, and neither party by virtue of this Agreement shall have any right, power or authority to act or create any obligation, express or implied, on behalf of the other party.

5. DevelopmentStaff-Monitoring.

A. Developer will utilize employees and/or contractors capable of designing and implementing the Software to be developed hereunder. All work shall be performed in a professional and workmanlike manner. Developer shall arrange for such employees and/or contractors, if any, to execute and deliver any document or instrument reasonably requested by Buyer to reflect Buyer's ownership of the Software or in connection with any application for patent or copyright.

B. Buyer shall have the right to reasonably observe and monitor all aspects of the performance by Developer of its obligations hereunder and Developer shall use reasonable efforts to facilitate such observation and monitoring. Information, functions and operations of Developer not directly related to its obligations hereunder shall not be subject to observation and monitoring.

6. Confidentiality.

A. Developer acknowledges that all material and information supplied by Buyer which has or will come into Developer's possession or knowledge of Developer in connection with its performance hereunder, is to be considered Buyer's confidential and proprietary information (the "Confidential Information"). By way of illustration, but not as a limitation, Confidential Information includes the Software, trade secrets, processes, data, know- how, program codes, documentation, flowcharts, algorithms, marketing plans, forecasts, unpublished financial statements, budgets, licenses, prices, costs, and employee and customer lists. Developer's undertakings and obligations under this Section will not apply, however, to any Confidential Information which: (i) is or becomes generally known to the public through no action on Developer's part, (ii) is generally disclosed to third parties by Buyer without restriction on such third parties, or (iii) is approved for release by written authorization of Buyer. Upon termination of this Agreement or at any other time upon request, Developer will promptly deliver to Buyer all notes, memoranda, notebooks, drawings, records, reports, files, documented source codes and other documents (and all copies or reproductions of such materials) in its possession or under its control, whether prepared by Developer or others, which contain Confidential Information. Developer acknowledges that Confidential Information is the sole property of Buyer. Developer agrees that disclosure of such information to, or use by, third parties, either during or after this Agreement, will cause Buyer irreparable damage. Developer agrees to use best efforts to hold Confidential Information in the strictest confidence, not to make use of it other than for the performance of its obligations hereunder, to release it only to the Developer's employees or contractors with a need to know such information and not to release or disclose it to any other party. Developer further agrees not to release such information to any employee or contractor who has not signed a written agreement between Developer and the employee expressly binding the employee not to use or disclose the Confidential Information, except as expressly permitted herein. Buyer shall be listed as a third-party beneficiary of any such agreement. Developer will notify Buyer in writing of any circumstances within its knowledge relating to any unauthorized possession, use, or knowledge of such Confidential Information. At any time, upon request, the Developer will return any such information within its possession to Buyer.

B. Developer acknowledges that Buyer's purpose in pursuing the development of the Software is to gain a significant competitive advantage over competitors operating without such Software and that such advantage will be jeopardized if such competitors learn of Buyer's negotiations with Developer or the performance by Developer of its obligations hereunder. Accordingly, Developer agrees to keep such negotiations and performance of its obligations hereunder strictly confidential and not to disclose any information to any third party or entity without the prior written permission of Buyer. In no event, shall Developer or any of its employees use Buyer as a reference in marketing Developer's services to any third party or entity without Buyer's prior written permission.

7. Training.

Developer shall provide Buyer and its employees with training consultations with respect to the use of the Software as may reasonably be requested by Buyer from time to time for __2 weeks__ [Time Training Available] after acceptance at no additional costs to Buyer ("Training Period"). Developer shall deliver a detailed user's manual to Buyer on or before completion of acceptance that will enable Buyer's employees who are otherwise unfamiliar with the Software to become adequately informed about using the software. All training that Developer is required to provide hereunder shall be performed at such locations and at such times as are mutually agreed to by the parties hereto. Upon the expiration of the Training Period and following Buyer's request, Developer will provide any support services necessary to insure Buyer's continued use of the Software. Such services will be performed on a time and material basis at Developer’s then current hourly rate for such services.

8. Warranties.

A. Developer warrants that for a period of ___1 month___ [Warranty Period] following acceptance, the Software will operate substantially according to the Specifications. In the event of any breach of the warranty in this Section 9. A., in addition to any other remedy to which Buyer may be entitled, Developer shall take all action necessary at its expense to cause the Software to operate according to the warranty.

B. Developer warrants that the Software will not infringe upon any copyright, patent, trade secret or other intellectual property interest of any third party. Developer will indemnify and hold Buyer harmless from and against all such infringement claims, losses, suits and damages including, but not limited to, attorney's fees and costs, and shall promptly following any bona-fide claim of infringement correct the Software so as not to be infringing, or secure at its own expense the right of Buyer to use the Software without infringement.

9. Term and Termination.

A. This Agreement shall commence upon the effective date and continue until all of the obligations of the parties have been performed or until earlier terminated as provided herein.

B. Developer's appointment as consultant pursuant to this Agreement and this Agreement shall terminate upon the occurrence of any of the following events:

(i) In the event either party defaults in any material obligation owed to the other party pursuant to this Agreement, then this Agreement may be terminated if the default is not cured following at least thirty (30) days' written notice to the defaulting party.

(ii) Either party is bankrupt or insolvent, or bankruptcy or insolvency proceedings are instituted against a party and the proceeding is not dismissed within sixty (60) days after commencement.

(iii) ___Victor Ciobanu_______ [Name of Principal Developer or Developers] dies or becomes disabled.

C. Section 2, Ownership of Software, and Section 7, Confidentiality, shall survive the expiration or termination of this Agreement. In the event of early termination due to Developer's default or the death or disability of the individual(s) identified in subsection B. (iii). above. Developer agrees to deliver the Software then completed.

D. If the Agreement is terminated due to the death or disability of Developer, then Developer (or Developer's executor, administrator or other representative) shall deliver that part of the Software then completed, provided payment is made by Buyer for such completed part.

10. No Waiver.

The failure of a party to require strict performance of any provision of this Agreement by the other, or the forbearance to exercise any right or remedy, shall not be construed as a waiver by such party of any such right or remedy or preclude any other or further exercise thereof or the exercise of any other right or remedy.

11. Assignment.

The rights, duties and privileges of Developer shall not be transferred or assigned by it, in whole or in part, without the prior written consent of Buyer.

12. Entire Agreement.

This Agreement constitutes the entire agreement between parties as to the subject matter hereof and supersedes all prior understandings or agreements whether oral or written. This Agreement may be modified only be written instrument signed by the parties hereto.

13. Successors.

This Agreement shall be binding upon and insure to the benefit of the successors and permitted assigns of the parties hereto.

14. Severability.

If any term of this Agreement is held by a court of competent jurisdiction to be invalid or unenforceable, then this Agreement, including all of the remaining terms, will remain in full force and effect as if such invalid or unenforceable term had never been included.

15. Governing Law.

The terms of this Agreement shall be construed and enforced under the laws of the State of New Jersey.

16. Headings.

The headings used in this Agreement are for convenience only and are not to be used in construction or interpretation.

IN WITNESS WHEREOF, the parties have executed this Agreement by their duly authorized representatives on the dates below written.

Zhong Li ______ Date July 16, 2021

Zhong Li, President of High Throughput Biology Inc.

Victor C ______ Date July 16, 2021

Victor Ciobanu

Exhibit A. SPECIFICATIONS/SCOPE OF WORK

Description of "Services":
Developer will develop a NFT-backed digital art marketplace according to the requirements specified in Exhibit B ("Requirements for an Art Marketplace Website"). Three milestones will be paid once requirements coded by corresponding highlighted colors as in Exhibit B are implemented.

- Milestone #1: $1100 upon delivery of functional code and installation instruction according to Exhibit B
- Milestone #2: $400 upon delivery of functional code and installation instruction according to Exhibit B
- Milestone #3: $500 upon delivery of functional code and installation instruction according to Exhibit B

Project "Deliverables":
All source code for a functional website that satisfies all requirements specified in Exhibit B and is accepted by Buyer. All documentation including instruction on how to set up and use the website. Deliverables need to be delivered before each milestone is paid.

Special Note: This is a fixed-fee project. Buyer agrees to notify Developer of any proposed changes in tasks or Deliverables, including any logically related assumptions and dependencies during the project.

Exhibit B: Requirements for an Art Marketplace Website

Introduction

This is an art gallery and marketplace for digital art. Digital artists can upload their digital art ("asset" in the rest of the document) for display and for sale. All conventional components of an ecommerce website should be implemented. NFTs will be created for uploaded assets and stored in a blockchain such as BSC. Uploaded assets are encrypted then stored in decentralized storage such as IPFS. An asset can be purchased with fiat currency using credit card or cryptocurrency using digital wallet. Transactions are captured in the blockchain.

Color Coding

Green: Milestone #1

Turquoise: Milestone #2

Yellow: Milestone #3

Graphic Design of the Website

A logo will be created for the website once the website name is determined. The website look-and-feel should be slick, welcoming, and tasteful. At least three different designs should be proposed and selected from. Different categories of art may have different themes.

Account Management

Standard online account management needs to be implemented. There are two types of account: admin and registered user.

Standard website admin portal will be built with functionalities such as managing layout, setting the fields for asset metadata, resetting user password, two-factor authentication, screening for offensive content, user support/ticketing, tracking transaction and browsing, etc. The admin portal also set the percentage the marketplace takes from each transaction. The admin needs to have a digital wallet.

Each registered user account should have a digital wallet. The digital wallet should hold a public/private key pair with sufficient strength. The public key should be stored in local database, while the private key is only stored in the user's digital wallet. Optional credit card information storage should be offered. The following information needs to be captured and maintained and displayed:

- Username/password and two-factor authentication
- User profile
- Digital wallet content after logging into digital wallet
- Uploaded assets as copyright owner
- Purchased assets
- Transaction history ordered by date

Asset Management

Asset Upload

A user can upload an asset once logged in. The user needs to acknowledge in a pop-up that s/he is the copyright owner of the uploaded asset. That confirmation should be timestamped and stored in a database. The user needs to provide metadata about the asset such as asset name, copyright owner/creator name, date of creation, timestamp of copyright ownership confirmation (auto provided by the system), description, art categories, keyword tags, etc. The fields of metadata can be updated by admin using the admin portal.

The uploaded asset will be processed by a server-side program A (outside of the scope of this project, a dummy program can be used). The input of the server-side program A is the uploaded asset and the user's public key for encryption. The output of the server-side program is a processed asset file (named "Processed Asset #1"), a second processed asset file (named "Processed Asset #2"), and a string. Errors produced by the server-side program A needs to be displayed. Processed Asset #1 is stored in database. Processed Asset #2 is stored in IPFS. A NFT is created with the URI of the IPFS location for Processed Asset #2 in the NFT's metadata. Other metadata for the NFT include name, NFT type, owner wallet address, description, the string from server-side program A, etc. The NFT is added to a blockchain and to the digital wallet of the owner. The owner needs to pay for the corresponding gas fee before the NFT is added to blockchain.

Asset organization

Processed Asset #1 is displayed. Assets should be organized according to categories or tags that they belong to. Assets can be sorted by likes, price, upload date, or other criteria. Selected assets should be featured according to criteria such as click rate, sale price, etc. The metadata for each asset along with the digital image/video/audio should be shown artfully. Comments and likes from registered users should be shown. Admin should be allowed to remove offensive comments.

Each uploaded asset may have zero or more copies backed by NFT, either for sale or not for sale. For each copy of an asset, its current owner should be listed, including records of previous sales and previous owners if available. The sale price for each copy of an asset should be specified by the current owner and listed in the fiat currencies or in cybercurrencies or both.

A collection is made available for each registered user to store the copies of assets s/he owns. The collection can be shown with or without hierarchy. If an owner wants to sell a copy of asset s/he owns, the owner clicks a button and choose a price then the copy is available for sale at the asset's page and is listed among all other copies of the same asset.

Initial Asset Tokenization for Sale

The copyright owner of an asset can decide how many copies of the asset s/he wants to sell by updating a counter for an asset and setting the price for unsold copies. The copyright owner can update price of unsold copies at any time. The prices of sold copies are controlled by their corresponding owners. The counter can be updated anytime and should be equal or larger than the number of copies sold. For each new copy created by the copyright owner, the following process is executed:

For each copy for sale, the corresponding Processed Asset #2 will be processed by the server-side program B (outside of the scope of this project). The input of the server-side program B is Processed

Asset #2 and the public/private key pairs of the owner. The owner will be asked to provide the private key since it is not stored in database. The output of the server-side program B is a processed asset file (named “Processed Asset #3”), a second processed asset file (named “Processed Asset #4”), and a string. Errors produced by the server-side program B needs to be displayed. Processed Asset #3 is stored in database. Processed Asset #4 is stored in IPFS. A NFT is created with the URI of the IPFS location in the NFT’s metadata. Other metadata for the NFT include name, NFT type, NFT ID of the NFT for Processed Asset #2, owner wallet address, price, description, the string from server-side program B, etc. The NFT is added to a blockchain and to the digital wallet of the owner. The owner needs to pay for the corresponding gas fee before the NFT is added to blockchain. The smart contract used to generate NFT should also specify the loyalty rate of subsequent sale to be passed to the copyright owner. The owner can choose the loyalty rate for each copy. The loyalty rate will be displayed for each copy.

A non-copyright buyer of a copy of an asset can be the seller of the copy by making a selection on his/her collection page.

Initial Asset Transaction

A NFT is minted after a buyer makes a payment to purchase a specific copy of an asset. The corresponding Processed Asset #3 is processed by a server-side program C (outside of the scope of this project). The input of the server-side program C is Processed Asset #3 and the public key of the buyer. The output of the server-side program C is a processed asset file (named “Processed Asset #5”) and a string. Errors produced by the server-side program C needs to be displayed. The Processed Asset #5 is stored in IPFS. A NFT is created with the URI of the IPFS location in the NFT’s metadata. Other metadata for the NFT include name, NFT type, NFT ID of the NFT for Processed Asset #4, owner wallet address, description, the string from the server-side program C output, etc. The NFT is added to a blockchain and to the digital wallet of the buyer. Processed Asset #3 is deleted from the database. The buyer needs to pay for the corresponding gas fee.

Payments to the seller of the specific copy of the asset, the copyright owner, and to the marketplace are made. The transaction is added to a blockchain. The ownership information on the copy of the asset is updated for the asset on the asset page.

Resale Transaction

The current owner of a NFT-based copy of an asset (not the copyright owner) can make the copy available for purchase by another registered user. An explicit action is needed since after a copy of an asset is purchased, the default is that it is not available for resale. After the owner clicks a button, s/he is prompted to enter the private key for decryption. The corresponding Processed Asset #5 is processed by a server-side program D (outside of the scope of this project). The input of the server-side program D is the Processed Asset #5, the private key of the buyer, and the string stored in the corresponding NFT’s metadata. The output of the server-side program D is a processed asset file (named “Processed Asset #6”). Processed Asset #6 is stored in database. Errors produced by the server-side program D needs to be displayed.

A NFT is minted after a buyer makes a payment to the seller, the copyright owner, and the marketplace to purchase the copy of an asset that is now available for purchase. The corresponding Processed Asset #6 is processed by a server-side program E (outside of the scope of this project). The input of the server-side program E is Processed Asset #6 and the public key of the buyer. The output of the server-side program E is a processed asset file (named “Processed Asset #5”) and a string. Errors produced by the server-side program E needs to be displayed. Processed Asset #5 is stored in IPFS. A NFT is created with the URI of the IPFS location in the NFT’s metadata. Other metadata for the NFT include name, NFT type, NFT ID of the NFT for the previous Processed Asset #5, owner wallet address, description, the string, etc. The NFT is added to a blockchain and to the digital wallet of the buyer. Processed Asset #6 is deleted from the database. The buyer needs to pay for the corresponding gas fee.

Payments to the seller of the specific copy of the asset, the copyright owner, and to the marketplace are made. The transaction is added to a blockchain. The ownership information on the copy of the asset is updated for the asset on the asset page.

Auction

The owner of a copy of an asset that is backed by a NFT is allowed to auction the copy at the website. The website should list all available auctions and allow sorting by time, price, category, etc.

Asset Download

The copyright owner can download Processed Asset #1 and Processed Asset #3 (if available) from database with corresponding links for download.

The current owner of a NFT will be given the option to download the asset represented by the NFT. Corresponding processed files will be retrieved from IPFS according to the NFT and sent it to a server-side program F (outside of the scope of this project). The server-side program F takes retrieved file and the owner’s public/private keypair as inputs. The owner will be asked to provide the private key since it is not stored in database. The output of the server-side program is a processed asset file. Message or error produced by the server-side program F needs to be displayed. The processed asset file can then be downloaded.

Asset Evaluation

Registered users can “like” an asset and provide short comments on an asset.

Shopping Cart and Transaction

A shopping cart is needed to facilitate asset purchase. The purchase can be made using credit cards based on multiple fiat currencies and using digital wallet for multiple popular cybercurrencies. Transaction history is added to blockchain and stored in database.