

# Tegi AI Marketplace

Technical White Paper  
July 2022

Jeremy Wurbs

## Collaborative AI Development, for Everyone

Tegi is an AI Marketplace for creating collaborative AI development organizations. In a traditional company, AI or other, a board determines the management decisions for the company, beholden to directives given by owners of that company. In virtually all cases, these directives will be to align the company with generating as much wealth as possible for these individuals. In a Tegi Collaborative,

- public ownership may not be sold or bought;
- development is shared and determined by all members of the collaborative;
- revenue is split as a profit-share within the collaborative;
- complete transparency is granted to a governing board, which reports to—
- a non-profit (governmental or other) who is responsible for legal governance;
- AI ownership transitions to a public entity over time (i.e. hastened copyright);

Note that a Tegi Collaborative is *not* necessarily a public entity; both development and usage rights may be either public or private.

The members of a Tegi Collaborative are individuals and companies who are generally expected to be for-profit entities. At its outset the collaborative founders determine the type of AI that will be developed by the collaborative and what rewards (monetary, ownership or reputation) will be given out to participants who successfully develop individual components. Ownership and reputation in the collaborative, including AI ownership and usage rights as well as rights to its current and future profits, cannot be sold or taken outside of the collaborative itself. That is, a Tegi Collaborative consists of for-profit companies and individuals creating AI for (generally) for-profit use-cases, where the ownership of the AI resides within the collaborative itself, transitioning to the public domain over time.

The motivation for creating the Tegi Collaborative, and the root for our belief in its necessity for AI development, is in recognizing that not all technological development is appropriate for maximizing the profits of an ownership class. A Tegi collaborative places limits on the profits that may be extracted from any singular AI development, while providing direct oversight to a governing agency along with transferring ownership to the public in a responsible, timely manner. Moreover, we recognize that some AI will be dangerous if released directly to the public, and the collaborative development described allows for potentially dangerous AI to be

siloed and contained, while still allowing the public to use (and profit from) the AI while governing bodies track and maintain kill-switch control over any individual AI on the platform long-term.

It is our hope that setting up AI development in this way, to be collaborative and inclusive, while providing transparent mechanisms for democratic governance and ownership, will lead to a better, more mature AI industry that works for all people, not just an ownership class at the top.

## Tegi Brief

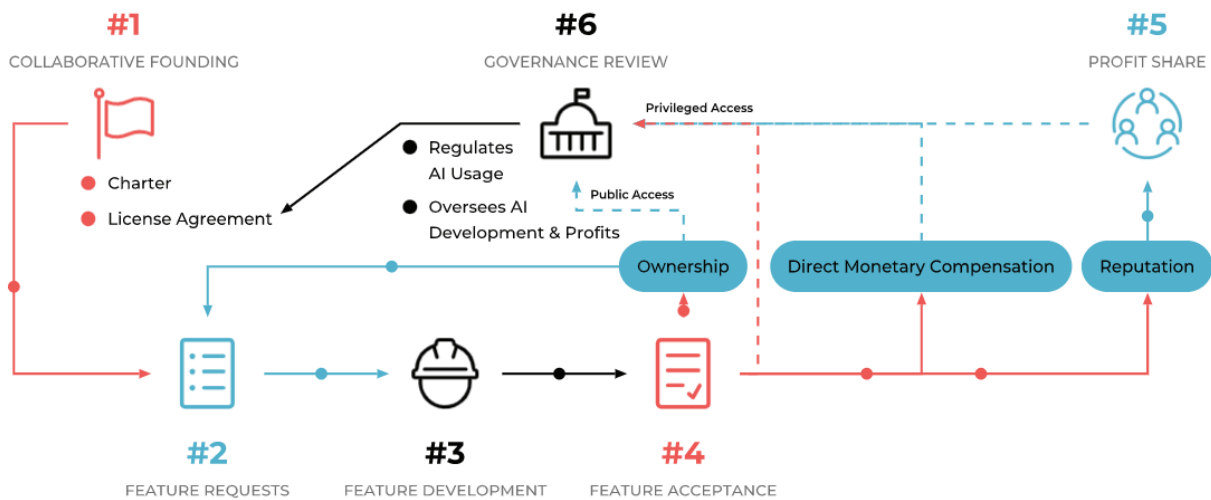
The Tegi Marketplace is loosely modeled after Colony<sup>1</sup>, a blockchain network for creating decentralized organizations. Colony seeks to address how decentralized work organizations can set development directions, assign ownership and credit, determine a single budget with revenue and expenditures, and (potentially) spawn sub-organizations around new tasks with a new set of participants. In the Colony metaphor, the AI Marketplace itself is the *Colony*, Tegi (the for-profit company) the *Root*, and each Collaborative its own *Domain*. Tegi does not itself own any part of any other collaborative (other than through direct participation in that collaborative as per any other member). Instead, Tegi offers the following guidance and support, in addition to setting the overarching ethos that defines the marketplace as a whole:

1. **Marketplace Platform.** Tegi develops and maintains the marketplace itself.
2. **Open-source Codebase.** Tegi develops and curates an open-source codebase for developers to use any AI on the marketplace, as well as adding their own AI or service APIs.
3. **Governance Support.** Tegi works with governing bodies to conduct research on the effects of AI in society, and implement any required regulations.
4. **Cloud Infrastructure.** Tegi provides push-button ready infrastructure to test and deploy AI on-demand, at scale. *Full disclosure: this is largely how we expect Tegi to make money.*
5. **Custom Consultation.** Tegi provides consultation services to help integrate first time clients onto the platform.

AI development itself is expected to occur within distinct collaboratives on the marketplace. A single collaborative is focused on a single AI model or product, which is defined and optionally shared through the collaborative through a single git repository. If additional models or software components are necessary, sub-collaboratives are created and linked directly back to their parent.

---

<sup>1</sup> Colony [white paper](#).



Broadly speaking, the expected order of operations for collaborative development on the Tegi platform is as follows:

1. **Collaborative Creation.** Any user may open a new Tegi Collaborative by providing:
  - a. **Collaborative Charter.** Document detailing the purpose of the collaborative, how membership is granted, and how ownership and profits are shared.
  - b. **License Agreement.** Document detailing how and by whom the created AI may be used. A standard license agreement is enforced by Tegi, in concert with associated governing agencies. Additional terms may be appended by the collaborative founder.
2. **Feature Requests.** Development is steered through well-defined feature requests. Any member with sufficient ownership or reputation may create feature requests. Owners vote to accept requests and determine any associated rewards.
3. **Feature Development.** Developers implement relevant features. For large feature requests, development itself may be paid to one or more specific developers.
4. **Feature Acceptance.** When a feature is complete, a pull request is made and the owners vote whether to accept or reject the request. If accepted, rewards are immediately paid to the developers.
5. **Profit-Share.** Most developed AI is expected to run under a SaaS model, continuing to pay out profit to developers based on earned reputation.
6. **Governance Review.** Viewing access is granted to governing oversight committees to help track the use and development of AI on the marketplace.

The rest of this document describes the overarching design of the Tegi Marketplace, focusing on AI development and profit alignment between corporates (both public and private) and governing agencies. At this time the framework described here should be taken as a general scaffolding, onto which further discussions and experience will be necessary to help refine the implementation details of creating an actual public-private AI marketplace ecosystem.

<b>Tegi AI Marketplace</b>	<b>1</b>
<b>Collaborative AI Development, for Everyone</b>	<b>1</b>
<b>Tegi Brief</b>	<b>2</b>
<b>Core Vision</b>	<b>6</b>
Current State of AI Development	6
Path to AI Monopolization	7
The Inevitable Release of Dangerous AI	8
Summary of AI Development Today	9
A New Vision for Collaborative AI Development	10
<b>Structure of the AI Marketplace</b>	<b>11</b>
The Colony	12
The Collaboration: One Domain, One AI, One Repo	13
Tasks	14
<b>Development Pipeline</b>	<b>15</b>
1. Create Collaboration	15
2. Grant Membership	15
3. Create Feature Requests.	16
4. Feature Development.	17
5. Pull Request.	18
6. Process Payment.	18
7. Request Arbitration.	18
<b>Tegi for Developers</b>	<b>19</b>
<b>Tegi for Consumers</b>	<b>20</b>
<b>Market Comparison</b>	<b>22</b>
<b>Appendix I: Current State of the Marketplace</b>	<b>24</b>
Home Page	26
Creating a New Collaboration	27
Subscribing to a Collaboration	30
Uploading AI to a Collaboration	31
Launching a New AI Service (Endpoint)	32
Using AI Services	35

# Why Tegi?

## 1 | Core Problems

- Consumers can't get custom AI
- AI Providers can't get paid
- AI set to dramatically increase wealth disparity
- AI is currently siloed & ungovernable

## 2 | Resulting Current Environment

- Fractured
- Expensive
- Dangerous

## 3 | Current Path

- Monopolization through acquisition
- Few, large players own & control AI
- AI built for financial gain, governed post release
- Loss of control over our shared AI destiny

## 4 | Tegi AI Marketplace

- Safe, Collaborative AI Development
- Shared AI Ownership
- Democratic AI Governance

## Core Vision

### Current State of AI Development

In 2012, an AI model called AlexNet won the ImageNet image classification competition— not only marking the first time a neural network had won a significant (read *difficult*), real-world machine learning challenge, but moreover winning by a *large* margin. The significance of this moment was recognized by scientists in both academia and industry almost immediately, and is generally credited with directly initiating the current Third Wave of AI. Over the intervening decade this Third Wave has turned into a full-blown AI Revolution.

Due to the preceding AI winter, lasting from the early to mid 1990s all the way until Alexnet in 2012, AI development, and in particular the deep learning algorithms driving AI today, was in 2012 only functionally known to a relatively small number of active researchers in academic institutions and departments. Many of the leading academic laboratories developing AI were bought in the initial years following the release of AlexNet. The AI forefathers and Turing Award winners Geoffrey Hinton and Yann LeCun, for example, both moved to lead Google and Facebook AI development, respectively, alongside many of their former students and laboratory members.

One fortuitous result of such widespread, rapid AI adoption by Big Tech, specifically through the purchase of highly specialized, otherwise hard-to-find university researchers, has been an academic camaraderie exceedingly rare in industry. Much, if not most, research from AI teams in Google and Facebook are published within months of completion. Often publications include

the actual code, and sometimes even include pre-trained models that could only reasonably be trained on large server farms, generally only available within Big Tech. Furthermore, it is not rare to see publications share authors from companies that are otherwise direct competitors.

This camaraderie has enabled AI collaboration to create and share AI knowledge across large, often competing parties in both academia and industry, and helped to democratize the creation of AI for a wide range of an ever-growing set of specialized AI developers. The main packages to develop, train and serve AI today– Tensorflow and Pytorch– for example, are open-source packages built, supported and maintained by Google and Facebook themselves, respectively.

So while this document outlines a vision for a better future for humanity, one in which AI is developed collaboratively, safely, with humanity's long-term interests in mind, it should be noted that the current state of AI development could be much worse, and that there is still a very valuable academic fervor pervasive in industry that genuinely believes in using AI for the betterment of society, to be as inclusive as possible, and to the extent that industry allows it, to resist the urge for large corporations to create and keep AI secret and siloed.

That is, it is not the intent of this document to label the leaders or participants of the AI industry as evil or malicious, but rather to work with those same participants, within the confines of industry as it exists in the real world, to build a better, safer ecosystem for AI development.

**It should be noted, however, that there are significant problems with how AI is currently developed in industry.**

The AI developed by Big Tech is owned by Big Tech. And Big Tech is both governed by the interests of shareholders and innately monopolistic. While a vibrant AI startup scene sprouted with the sudden interest in AI after the release of Alexnet, the majority of successful AI startups are venture backed and looking to IPO or get acquired by Big Tech– i.e. in virtually all cases AI startups are running deficits to create valuable AI technology to sell-out to an owner class looking to directly, and quickly, make back their initial investment, plus more.

This deficit-driven development cycle is driven by the functional requirements of the real-world and largely inescapable by most AI startups today–

## Path to AI Monopolization

Most commercial AI uses require significant customization to the use case at hand. As AI technology is still new, however, most would-be AI consumers do not have the in-house development teams with the expertise required to build, or even support, their own AI solutions. Would-be AI consumers, then, must rely on AI development partners to provide and service them with end-to-end, push-button ready solutions.

But AI providers, often startups or small teams in academic institutions, do not have the internal resources to build out large software development teams and packages, create the (cloud)

infrastructure necessary to serve their AI, or conduct lengthy proof-of-concept deals to tailor their technology to potential customers before a license deal is reached.

The result is often that a small, two to eight person team of specialized AI researchers are able to create and prove-out promising AI technology, but not have the resources to serve them to clients at scale. The team is forced to use the promise of their technology to raise money from VC firms in order to build out an internal team of software engineers to develop end-to-end product solutions that can be delivered and served to clients, at scale.

During this development period the AI startup will be looking to secure multiple meaningful license deals, as a single deal is rarely enough to cover development costs, which requires expanding the team further and subsequently running larger deficits as the team size grows. Ideally the company reaches an inflection point and is able to turn a profit, but generally the goal for such a startup is not even to become profitable, per se, but rather to maximize their valuation and exit via IPO or acquisition in order to satisfy the now-significant set of shareholders to whom years of development costs are owed— shareholders in debt, looking to make back their investment, plus more, as quickly as possible.

That is, almost invariably, for small AI teams to be successful today, they will need to use VC funds to build out the infrastructure to be commercially successful, acquiring stakeholders in the process who are innately interested in steering the company towards maximizing profits and return on investment. Once a successful startup owes a large debt to a set of shareholders, the ones who will end up owning the associated AI technology and IP will be the ones who are willing to pay those shareholders out— Big Tech.

The result is a system wherein AI ownership ends up in the hands of shareholders looking to maximize profits to get a return on investment, and where long-term AI development and ownership is cornered into an increasingly smaller number of monopolizing Big Tech companies looking to maximize their slice of the AI ownership pie.

## The Inevitable Release of Dangerous AI

While the monopolization of industries is a common thread in capitalistic systems, in reference to AI, there are additional global risk management concerns that arise as well. Namely, some AI will invariably be *dangerous*.

As discussed, AI development currently follows old academic norms. That is, development occurs in secret, by siloed teams who do not divulge what they are working on until the publication is released— often coinciding with the release of the AI itself. Unfortunately, the race for first authorships (not to mention associated industry-driven profit motives), often dictates that no one in the silo asks if a technology *should* be developed— rather, the only questions passed down the hierarchy are *can*, *how fast*, and *how much*.

It is true that most, if not all, AI will be largely beneficial on release. But *some* AI will invariably be dangerous, at a point far *after* it is released. The first convolutional neural network, LeNet, was first published in 1989— twenty three years later a re-implementation of LeNet, the aforementioned AlexNet, launched the AI revolution. The operative difference between the two is data and compute capability— i.e. pieces that were inevitable. That is, in the case of AI technologies, the general policy is *release today, find out what was released over the following decades*.

For instance, the world did not ask for, nor even know about, AlphaFold before its release. And by and large, over the coming years and decades, AlphaFold and its successors will invariably prove instrumental in helping scientists researching and developing medical therapeutics that will save many lives. But, invariably, AlphaFold will also lower the barrier to creating novel, epidemic-level pathogens as well, and increase the number of people able to create a man-made pandemic in secret— not necessarily today, but, perhaps, twenty three years from its release.

There was no governmental oversight committee that gave the permission to release AlphaFold. No extensive study was conducted to examine what genetic engineering technologies may be directly unlocked by its release. But here it is, for better or worse.

## Summary of AI Development Today

In short, AI development today is (i) fractured, with no single place to go to find AI collaborators, both consumers or producers, (ii) expensive, with development cycles requiring lengthy proof-of-concept projects and production scaling before revenue deals can be reached, (iii) profit-driven, driving monopolization of the AI industry and dramatically increasing wealth disparity, and (iv) dangerous, lacking any governance oversight mechanisms to manage what AI is developed and how they are released onto the world.

That is, concisely put, the problems sought to be directly addressed by the Tegi Marketplace are:

- Consumers cannot obtain custom AI easily;
- AI providers have no path to direct monetization;
- AI development is increasingly beholden to a shareholder class, monopolizing the industry and dramatically increasing wealth disparity;
- AI development is currently siloed and ungovernable;

The implicit thesis within this document is that without direct, purposeful course correction the current path of AI development will lead humanity in the wrong direction— towards a world where AI is used to raise an ownership class higher at the expense of the common person, where AI development will be increasingly more dangerous as effective oversight is not possible, and where we, humanity as a whole, have no effective agency to choose the future we will inhabit.



# What is a Tegi?

A new way to create, own, deploy and govern AI

## 1 | Core Vision

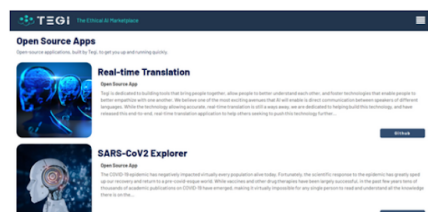
- Safe, Collaborative AI Development
- Shared AI Ownership
- Democratic AI Governance

## 2 | Definition of Success

- End-to-end AI Development
- Direct Monetization Path
- Ease of Collaboration
- Integrated Governance

## 3 | Solution: An AI Marketplace

That implements “success”



## A New Vision for Collaborative AI Development

Tegi represents a new way to create, own, deploy and govern AI.

The Tegi AI Marketplace is meant to serve as one tool in humanity’s collective AI toolbox to alleviate the AI development alignment problems described above. It is not expected that the marketplace structure described here would solve all AI development concerns or be capable of solving all potential client use cases. Still, it is the express goal of the marketplace, as described below, to help incorporate the following core values into AI development, writ large:

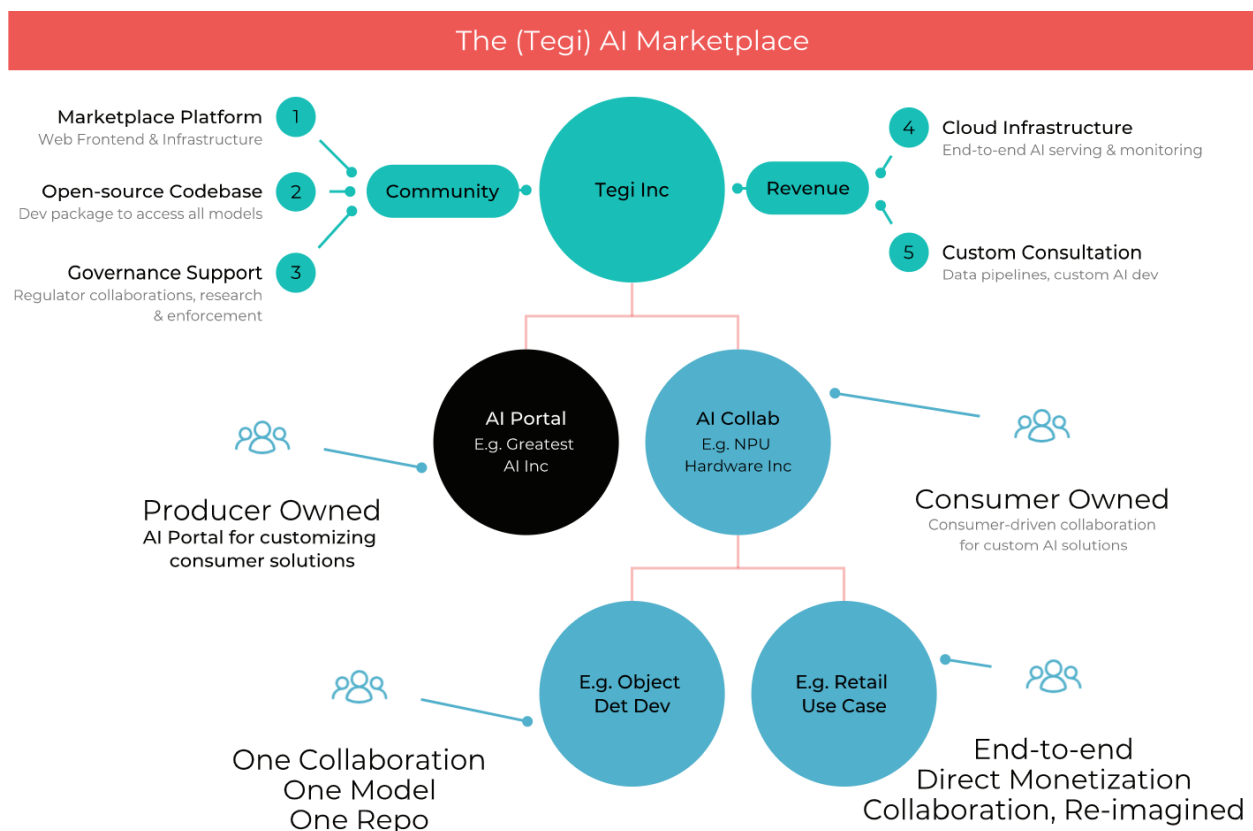
- Safe, Collaborative AI Development
- Shared AI Ownership
- Democratic AI Governance

As justification— desiring AI development to be *safe* is self-evident; desiring AI development to be *collaborative* enables individual, self-interested AI developers to join together to collectively build out custom solutions for AI consumers without requiring AI ownership or development transfer to Big Tech or an indebted shareholder class; desiring AI ownership to be *shared* means that a wider class of people will own the direction of AI development, minimizing wealth disparity as well as profit-driven development; *democratic governance* combines two concurrent goals that AI should be developed with direct, *governance* oversight, and that that governance should represent the *democratic* will of all people, not just those in the direct market-driven interaction, as all people will be affected by AI, whether or not they are direct participants in the marketplace.

Following, we identify the following explicit *success criteria* to help steer marketplace design and provide key, qualitatively measurable proxies for how well the marketplace is working towards enabling the core values identified above:

1. **End-to-end AI Development.** Are AI consumers able to create collaborations and obtain push-button ready AI solutions from the marketplace? Concurrently, are AI providers able to directly serve their developed AI technology through the marketplace without needing a large, internal software engineering team?
2. **Direct Monetization Path.** Is the marketplace able to provide direct license deals or SaaS revenue to AI developers?
3. **Ease of Collaboration.** How much time, money, resources and rate limiting steps is the marketplace able to reduce for market participants? Is the marketplace able to reach a critical point where small AI teams, even individuals, can serve AI to large clients at scale?
4. **Integrated Governance.** Are effective governing structures in place and able to offer effective control over our shared AI future? Is the marketplace able to help reduce, predict or limit the damage from dangerous AI development?

## Structure of the AI Marketplace



The Tegi Marketplace is loosely modeled after [Colony](#), a blockchain network for creating decentralized organizations. Colony seeks to address how decentralized work organizations can set development directions, assign ownership and credit, determine a single budget with revenue and expenditures, and (potentially) spawn sub-organizations around new tasks with a new set of participants. In the Colony metaphor, the AI Marketplace itself is the *Colony*, Tegi (the

for-profit company) the *Root*, and each Collaborative its own *Domain*. Tegi itself does not own any part of any other collaborative (other than through direct participation in that collaborative as per any other member). Instead, Tegi offers the following guidance and support, in addition to setting the overarching ethos that defines the marketplace as a whole:

1. **Marketplace Platform.** Tegi develops and maintains the marketplace itself.
2. **Open-source Codebase.** Tegi develops and curates an open-source codebase for developers to use any AI on the marketplace, as well as adding their own AI or service APIs.
3. **Governance Support.** Tegi works with governing bodies to conduct research on the effects of AI in society, and implement any required regulations.
4. **Cloud Infrastructure.** Tegi provides push-button ready infrastructure to test and deploy AI on-demand, at scale.
5. **Custom Consultation.** Tegi provides consultation services to help integrate first time clients onto the platform.

The structure of the marketplace is discussed next, starting with the fundamental concepts of ownership and reputation, moving to the hierarchy of collaborations, and then walking through the individual steps of developing pieces of work within a collaboration.

## The Colony

The Tegi Marketplace operates as a Colony composed of individual Distributed Asynchronous Organizations (DAO). The fundamental unit for each DAO is a *domain*, each of which contains its own, distinct set of users. Each domain constitutes a node within a tree hierarchy– the tree is, itself, the *colony*. In Tegi, each of these domains is further defined and known as a *collaboration*.

One of the most fundamental questions to answer in crowdsourced collaborations is how ownership of the developed IP is split between members. The Tegi Marketplace loosely models its ownership structure after one of Colony's key concepts– the separation between *ownership* and *reputation*.

Ownership, in a Tegi Collaborative, means having the right to determine the development priorities of the collaborative and who may use the resulting AI. Reputation, on the other hand, conveys the right to collect profits from the collaborative and its development efforts. In general, the founders and/or benefactors of a collaborative collect *ownership*, while developers who manage to create successful pull requests collect *reputation*.

Ownership does not decay by itself, and generally operates within a Tegi Collaborative as would be expected from the common use of the term. Reputation, however, does decay over time, and is used in Tegi similarly to its definition in Colony: *a non-fungible, time-decaying measure of aggregate past contributions*.

It is Tegi's express belief that society, writ large, should benefit from AI development, including sharing directly in its profits. Thus, Tegi collaborations add in a permanent reputation allotment earmarked for non-profit and government funding directly aligned with the collaboration at hand. That is, if the collaboration is developing AI to automate manufacturing processes, then a portion of the profit share should be set aside for funding non-profit and government organizations helping to oversee and take care of affected workers in that industry, as well as organizations providing leadership and regulations (whether legally binding or not) that guide how AI should be used in the manufacturing industry.

The explicit parameters defining reputation decay may, in part, be defined by the members in a collaboration, while some constraints, such as the percent that must be earmarked for the public share, may be dictated by governing bodies or by Tegi itself. Other Colony components co-opted into the Tegi structure are further detailed below, including *Tasks*, *Permissions*, and *Budgeting*.

One key requirement addressed by Colony *not* considered by Tegi is the blockchain itself. The reasoning here is twofold: (i) first, while collaborations may be private, Tegi users are, as a general principle, *not* anonymous, and (ii) second, Tegi, and any other root nodes in the marketplace, are expected to be large, *trustable* entities. As such, there is currently no design or plan to integrate any blockchain technology explicitly into the Tegi Marketplace, although this design choice might change if the ownership model implemented by Tegi were to expand to cover new constraints in the future.

## The Collaboration: One Domain, One AI, One Repo

As in Colony, Tegi bases its fundamental unit of collaboration around a domain. Unlike Colony, however, Tegi collaborations are generally expected to be focused solely around software development and, more specifically, AI development. As such, Tegi makes the design choice to *tie a single domain to a single git repository*. When multiple repositories are desired, sub-domains may be spawned and used for more complex organizational structures.

This design choice is not unique to Tegi and, in truth, is or has already become the general industry trend. That is, in the past, an AI model was commonly thought to be a single Tensorflow or PyTorch graph. More recently, however, as the field and associated toolchains have advanced, the term "AI model" is taken more broadly to be an entire code repository. For instance, models on the Hugging Face Model Hub, the current largest AI marketplace in the world, are all (custom) git repositories, which may themselves contain one or more Tensorflow or PyTorch graphs.

While potentially constraining, the rationale for tying a collaboration to a single repository, as opposed to allowing multiple repositories or packages, stems from the desire to balance two, often-competing goals: (i) provide a streamlined development pipeline that can be managed by non-developer clients, and (ii) provide a rich enough development environment so that multiple, often competing developers can work together to create proprietary AI customizations required by the end client.

Note that there is the implicit assumption that owners in Tegi domains (i.e. collaboration owners) are all aligned and mutually interchangeable. That is, they are either single individuals, multiple individuals who all work for the same company or organization, or are, in general, otherwise clearly aligned and mutually trustworthy. Developers, on the other hand, are expected to, in general, neither know each other nor otherwise have any means to coordinate development efforts outside of the structure provided by the collaboration platform itself. Moreover, depending on the collaboration structure, developers may have directly competing interests.

Thus, in an effort to strike an appropriate balance that enables (potentially-competing) developers to work together in a crowd-sourced environment while being managed by end-clients that may have limited working experience in AI technologies, Tegi has chosen to marry individual domains with a single (git-based, custom) repository structure. The collaboration workflow is examined in more detail in the [Development Pipeline](#) section.

## Tasks

In Colony, individual units of work are defined through an explicit *Task* structure. In a Colony task, a *manager* defines and coordinates delivery of the task, a *worker* is responsible for executing the task, and an *evaluator* is responsible for confirming that the work has been completed successfully. Additionally, tasks also have an associated *due date* and *payout*.

In Tegi, a task similarly defines a fundamental unit of work, with the simplest case specifically being a feature request within the shared repository. In the nominal case, any collaboration *owner* may act as the manager, curating a list of feature requests with associated due dates and payouts; *developers* act as the workers and check out tasks, either freely or by request, optionally into separate (forked) repositories, where development work is carried out; once development work is complete, a pull request is made to the *evaluator*, who asserts that the new code meets the requirements and passes the code back to the manager. In the ideal case, the collaboration owner is able to provide an automated mechanism, such as a test dataset with set performance criteria, to test that each feature request is successfully implemented.

Tasks themselves may be placed within any other project management system, such as kanban or scrum-based development cycles, in order to manage larger projects and goals. The key feature of a task is that it represents the smallest unit of meaningful work and always grants the developer(s) one or both (i) direct monetary compensation and (ii) reputation. In the case that multiple tasks must be completed together to form a functional unit, tasks may be *linked*. Linked tasks have additional payout bonuses that are given to all task providers iff all tasks in the linked set are completed.

# Development Pipeline

Step	Owner	Member
1	Create (Sub) Collaboration	
2	Invite Members to Join	Join / Request to Join
3	Create Feature Request	Propose Feature Request
4		Checkout / Fork Feature Branch
5		Complete Development Work
6		Make Pull Request
7	Accept / Reject / Request Amendment	
8	Process Payment (Monetary / Reputation)	Receive Payment (Monetary / Reputation)
9	Request Arbitration	Request Arbitration

For developers, using a git-based workflow will likely seem natural. Following, work in a collaboration between owners and members (developers) follows the steps outlined below.

## 1. Create Collaboration

Any user may create a collaboration by providing the following documents.

- A. **Collaborative Charter.** Document detailing the purpose of the collaborative, how membership is granted, and how ownership and profits are shared.
- B. **License Agreement.** Document detailing how and by whom the created AI may be used. A standard license agreement is required by Tegi, in concert with associated governing agencies. Additional terms may be appended by the collaborative founder.

## 2. Grant Membership

Users must be granted membership into a collaboration in order to participate. Collaboration, membership and user IDs across the Tegi are not anonymous. That is, collaboration charters, membership lists (owners and reputation holders), as well as user identities are not anonymous and can be viewed by anyone with viewing access. Collaborations can, however, set the following rights to public or private:

#### A. Visibility.

- a. *Public.* Anyone on the Tegi platform can view the collaboration and request to join. Viewing, in this case, specifically refers to being able to see the information provided upon collaboration creation, as well as public information on the collaboration owners and reputation holders.
- b. *Private.* Only users who are invited to join the collaboration can view the collaboration. It will not appear in any searches or directory results on the marketplace for non-members.

#### B. Membership.

- a. *Public.* Anyone is allowed to join and participate as a provider. Note: the collaboration visibility must be set to public in order to set the membership to public.
- b. *Private.* Users must request access to join, or otherwise be invited by the collaboration owners.

#### C. Usage.

- a. *Public.* Any *member* in the collaboration may use the produced AI.
- b. *Private.* Only *owners* of the collaboration may use the created AI.

### 3. Create Feature Requests.

Development in Tegi collaborations is driven by a series of *feature requests*. Each request is a well-defined, self-contained contract that specifies a set amount of work to be completed, along with acceptance criteria and payouts for any associated developers.

- A. **Problem Context.** If it is not readily obvious, the feature request must include contextual information so that developers understand what problem they are solving and how their solution will be used.
- B. **Description.** The feature request itself should be clear, self-contained and describe any AI that needs to be trained or created, including specific input-output data definitions and any resource constraints expected to be met.
- C. **Data.** If the feature request requires training an AI model, it must include data or clearly describe an acceptable data source.
- D. **Acceptance Criteria.** Clear, well-defined acceptance criteria must be stated. If the feature request involves developing AI where the model performance is one criteria of acceptance, a well-defined metric tied to an available dataset must be given.
- E. **Evaluation Tests.** While not strictly required, a proper feature request would include code to automatically run unit tests and validation on any proposed solution. This code may run on the Tegi cloud directly and be used to automatically accept pull requests obviating the need for a human evaluator.
- F. **Evaluator.** If the acceptance criteria cannot be verified autonomously, the feature request must include a non-anonymous evaluator to judge whether a pull request has successfully met the acceptance criteria or not. The evaluator must be a member of the collaboration, and may not submit a solution pull request to the feature request at hand.
- G. **Checkout Permissions.** Who is allowed to checkout the feature request and submit solution pull requests. In general, feature requests are allowed to be one of two types:

- a. *Open*. Open requests may be checked out by anyone at any time, and may be set to either (i) first-past-the-post or (ii) competition. First-past-the-post requests are paid out to the first solution that meets the acceptance criteria, while competitions must have a well-defined, measurable performance metric to judge solutions. Competitions maintain a leaderboard, and pay out to the highest performer on a set end date.
  - b. *Reserved*. Reserved requests may only be checked out upon request, requiring approval from a collaboration owner. Reserved requests will pay out to any submission meeting the acceptance criteria, even if multiple submissions are successful, although a set deadline may be included in the acceptance criteria.
- H. **Payouts**. Payouts may be one or both of (i) direct monetary compensation and (ii) reputation. Reputation may only be given if the collaboration has a means, even if only at a future time, to generate SaaS revenue.
- I. **Potential Solutions**. While not required, it is generally a good practice for the person creating the feature request to describe one or more potential solutions that they feel would solve the request. Doing so allows developers to better understand the requester's expectations and the scope of the work that may be required.

#### 4. Feature Development.

There are three main steps for feature development that are always followed: (i) checkout / fork a new feature branch, (ii) the development work itself, including code merger, and (iii) the final pull request. While these three steps will always be followed, depending on where the resulting IP is expected to be developed and owned, there are two different paths for handling feature development. Namely, public or private.

- C. **Public**. If the development (code and model weights) are expected to be open or public to the entire collaboration, and there are no other direct competing interests, then development may occur within the same repository and the developers simply checkout a new feature branch where they conduct the development. The pull request and merger happens normally, as with any other git-based repository, with payouts occurring upon acceptance.
- D. **Private**. If the development efforts are expected to be owned privately (either by the collaboration owners or developers), or if there are directly competing interests that preclude open development, then feature request development may fork the original collaboration repository into its own, new domain, effectively creating a sub-contract collaboration where the new collaboration owners are the developers in the parent domain. Development continues in this forked branch as normal, with two potential options for merging code back into the main repository when complete:
  - a. **Single Repo**. In the simple case, the forked branch is merged directly back into the parent repository. This situation may occur if the fork was used to protect the developers, assuring that they would receive payment before having to give up their development efforts, or in the case where developers are effectively competing against each other, and so do not want to give up code before it has been accepted.



- b. **Multiple Repo.** In the case where the code is not to be merged back into the parent repository, either for IP or technical reasons, the child repository will be uploaded onto the Tegi platform as its own collaboration, with usage permissions set in accordance with the parent collaboration's feature request. Once the fork has been accepted and paid, it will not be able to be deleted by the developers except by agreement with the parent collaboration owners, and will be served and made available to the parent collaboration through the Tegi cloud, as per any other AI.

## 5. Pull Request.

Once feature development is complete, the developer will make a pull request to push their code back to the main branch and to be used by the end client. The pull request opens a privileged dialogue open to the collaboration owner, feature developer and the evaluator. Ultimately the decision to accept or reject the request lies with the evaluator. There are three possibilities:

- A. **Accept.** In the simplest case, the submitted solution passes all acceptance criteria and may be accepted immediately. Accepted work is then pushed or made available to the collaboration's main branch.
- B. **Request Amendment.** If the development work does not match the acceptance criteria, but is otherwise of high quality and needing only reasonable adjustment, the evaluator may accept the solution upon future achievement of an amendment. In this case the evaluator must specify which acceptance criteria was not met and what change may be made to successfully meet that acceptance criteria.
- C. **Reject.** If the developed work does not match the acceptance criteria, and the evaluator judges that it would be better to start over than to continue the current development, they may reject the solution outright. Rejected work is deleted from the collaboration repository and may not be used by the collaboration owners or other members without express consent from the developer.

## 6. Process Payment.

Once development is accepted, any monetary rewards and reputation gain are deposited to the developer's account.

## 7. Request Arbitration.

Arbitration requests may be made to either the evaluator or Tegi. Note that Tegi is not responsible for determining whether any particular development effort meets the given acceptance criteria, which is the sole purview of the evaluator. Tegi may aid, however, in making sure that any developed code or AI runs smoothly on Tegi (cloud) resources.

## The Business Case

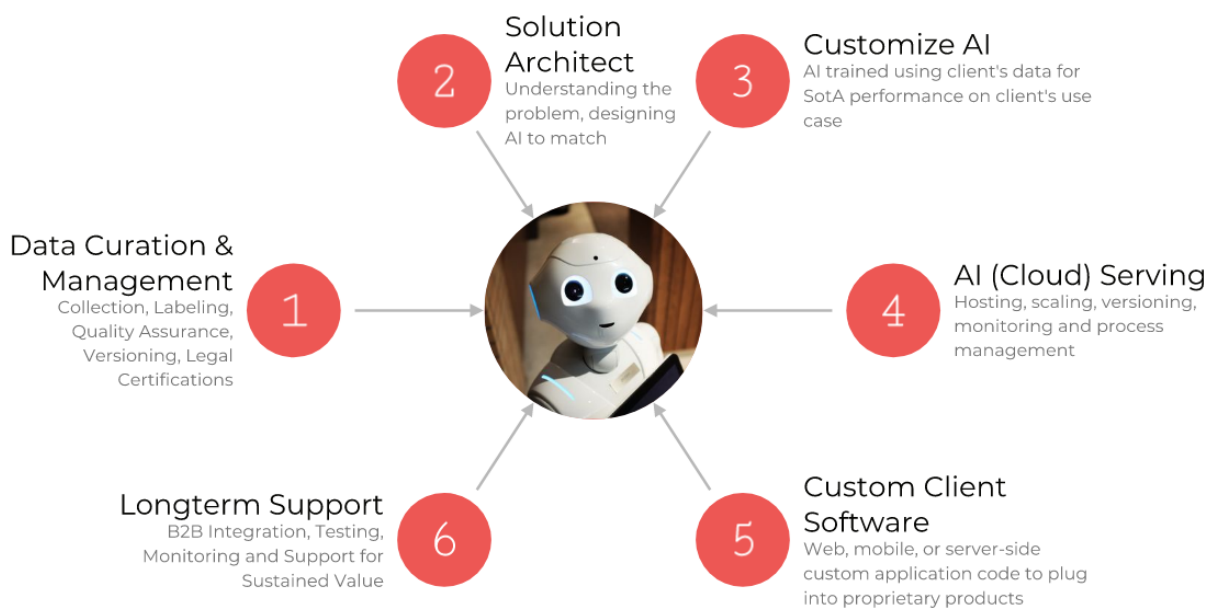
Marketplaces are becoming increasingly popular in the digital economy. These platforms connect buyers and sellers, and have become essential for e-commerce, peer-to-peer transactions, and a distributed work economy. The success of a marketplace, however, depends on more than just having a great idea or solid technology platform, whether that platform is Tegi or any other AI marketplace that seeks to enter the market. The marketplace must reach a tipping point where network effects can take place and make the marketplace self-sustaining.

We believe in the Tegi vision because of the necessity of a marketplace that serves humanity. In addition, however, no marketplace can be successful without assuring that the needs of both the consumers and suppliers (AI providers in this case) are sufficiently met. While not the focus of this whitepaper, in this section we make an argument that the Tegi marketplace is able to meet client needs while being easily integrated into developer pipelines.

## Tegi for Consumers

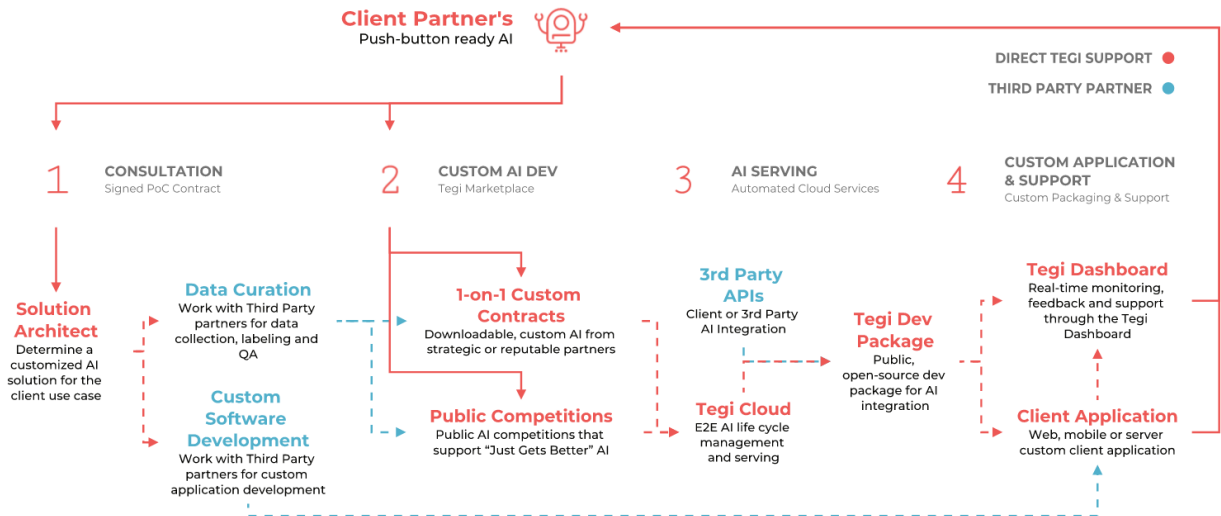
Artificial Intelligence (AI) has become an increasingly popular solution for businesses to improve operational efficiency and gain a competitive edge in their respective industries. However, the adoption of AI by consumers is a complex process that requires careful consideration and planning. One of the primary goals of the Tegi marketplace is that customers, both businesses and individuals, can manage the process of AI development, deployment and continued support entirely within the Tegi ecosystem.

Towards that end, there are six key steps that the platform seeks to help AI customers address when creating AI-based solutions.



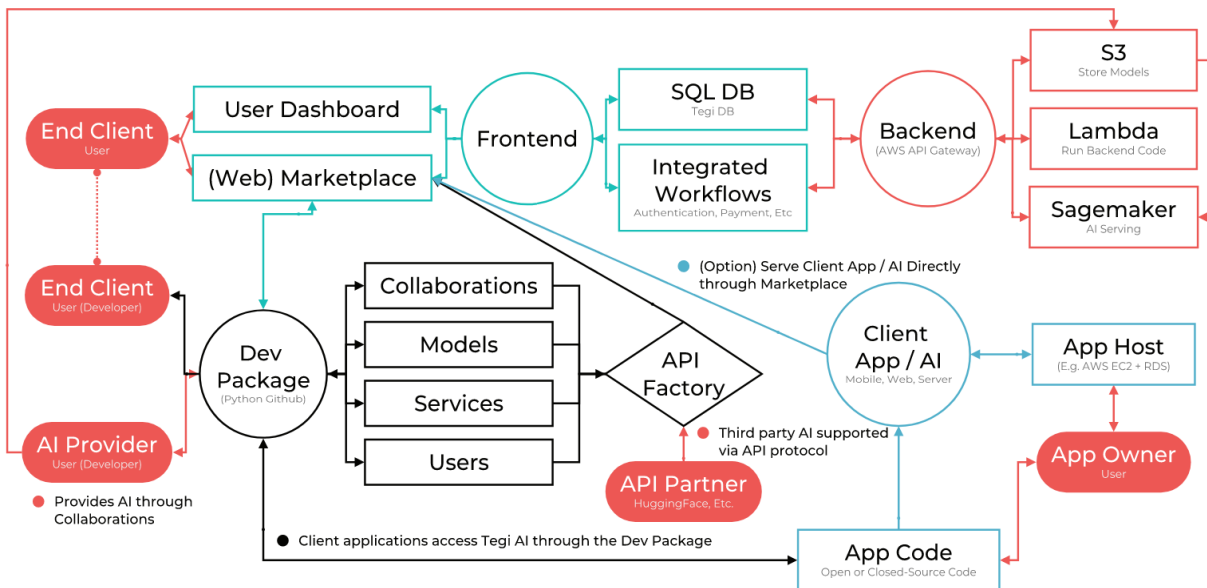
1. **Data Curation and Management.** One of the critical steps in using AI is data curation and management. This process involves collecting relevant data, labeling it, ensuring its quality, versioning, and obtaining the necessary legal certifications. Data curation and management are essential because AI models are only as good as the data they are trained on. Therefore, AI consumers must invest significant time and resources to ensure that they have the right data for their specific use case.
2. **Solution Design.** The next step in the AI adoption process is solution design, typically aided through a solution architect. This process involves understanding the problem that needs to be solved and designing an AI model that matches the use case. AI consumers must work closely with AI solution providers to develop a customized solution that meets their specific needs. This process requires a deep understanding of AI technologies and their capabilities to ensure that the solution is tailored to the client's unique requirements.
3. **AI Customization.** Once the AI solution has been designed, the next step is AI customization. This process involves training the AI model using the client's data to achieve state-of-the-art performance on the client's use case. AI consumers must provide high-quality data to the AI provider to ensure that the AI model is trained accurately. The AI model's accuracy and effectiveness depend on the quality and quantity of the data used for training.
4. **AI (Cloud) Serving.** After the AI model has been trained, it needs to be hosted, scaled, versioned, monitored, and managed. AI consumers can choose to host their AI model on the cloud or on-premises. Cloud hosting offers many benefits, including scalability, cost-effectiveness, and ease of management. AI consumers must choose an AI provider that offers reliable cloud hosting services and can manage the AI model effectively.
5. **Custom Client Software.** In many cases, AI consumers will need to build custom client software, including web, mobile, or server-side applications, to interact with the AI model. This process requires specialized skills and expertise to ensure that the software is compatible with the AI model and can handle large volumes of data. AI consumers must work closely with software developers to ensure that the software is optimized for the AI model.
6. **Long-term support.** Finally, AI consumers must plan for long-term support, including B2B integration, testing, monitoring, and support for sustained value. The AI model must be continuously monitored and updated to ensure that it remains effective and relevant. Additionally, AI consumers must plan for ongoing testing and monitoring to identify any issues or areas for improvement. AI providers must offer reliable support services to ensure that clients can achieve sustained value from their AI investment.

In order to successfully serve AI customers on the platform, it is expected that the Tegi marketplace meet all of the above requirements, potentially including the Tegi company directly for help in AI design. Subsequently, a default roadmap for how an AI customer may look to use the marketplace to develop, integrate and support AI into their proprietary applications is shown in the following diagram.



## Tegi for Developers

Tegi curates an open-source developer's package that can be used to incorporate any AI from the Tegi Marketplace into end-client applications. As part of the developer's package, Tegi also defines an API that can be used by third party AI providers to serve their AI through the Tegi Marketplace themselves. While providing detailed documentation on how to use this API is beyond the scope of the present document, the key components of this API are outlined below in order to provide context on how the marketplace is structured together and how third-party vendors may connect their AI solution suites to the open market.



The Tegi API defines the following key components of the platform:

1. **Users.** In order to use any AI through the Tegi Marketplace, it is required that a user login and authenticate every transaction. Every user, therefore, has an account on the marketplace that is connected to a payment gateway, through which they automatically pay for services. In order for third party vendors to support users on the Tegi platform, they must provide an authentication and payment gateway. Users may then subscribe to third party AI and services, which may be accessed directly through the Tegi web portal or developer's package.
2. **Collaborations.** In order to be directly deployable, collaborations need to define a set of *pipelines*, each of which is simply an input-output relationship on a piece of data. It is generally expected that collaborations further provide test data for each pipeline in order to test and rank different submitted AIs. In this case, any newly submitted AI will automatically be scored on the test data and placed in a collaboration leaderboard. In general, the best available AI model will automatically be deployed and used by the collaboration.
3. **Resources.** Resources are any compute node or resource that is capable of running an AI model. Resources have varying constraints, specs and prices, and may be supported by either Tegi or a third party vendor.
4. **Models.** Classically, models are understood to be AI technology implemented as PyTorch or Tensorflow graphs. While the expectation within the Tegi Marketplace is generally the same, for the Tegi API the implementation details of a model are not actually considered, but rather a model must, at minimum, subscribe to one or more collaboration pipelines that it supports as well as one or more resources that are capable of running the model.
5. **Services.** All AI on the Tegi marketplace must be launched as a *service* before it can be used. A service always subscribes to one or more collaboration pipelines when it is created and launched. There are two types of services, depending on the service provider:
  - a. *Tegi.* Any model on the Tegi marketplace may be turned into a service by launching it on a supported resource. These services are then immediately available to any user subscribed to the associated collaboration with the appropriate usage permissions.
  - b. *Third Party.* Outside AI vendors, running their own cloud services, need only support the User and Service API components. Third party services are still directly tied to collaboration pipelines, but need not use or support any other Tegi resources or models.

Taken together, we hope that the Tegi marketplace serves as a simple, one-stop solution for AI developers to integrate their technology into a broader marketplace and get fairly compensated, all while maintaining a connection to a broader democratic community voicing how the associated development of AI should continue.

# Market Comparison

One of the key reasons behind building Tegi is the belief that, currently, there are no other platforms on a path to help humanity build AI both democratically and responsibly. For better or worse, market forces dictate the relationship that businesses will have with AI, namely reaching for ever more powerful AI algorithms to solve problems for their business use cases. The monetary reward for incorporating AI into product pipelines is so great that it is unlikely a moratorium or cap on AI development will be possible. What we hope to foster with Tegi is a platform that enables responsible, continued development of AI that is able to customize AI for individual business use cases while retaining democratic control of our shared future.

We believe that the Tegi marketplace model will be successful as it is the only platform we are aware of that enables end-to-end AI development for customers, a direct monetization path for developers, and seeks to include integrated, democratic governance policies directly into a democratically developed AI ecosystem.

	Tegi	AWS	HuggingFace	Kaggle
End-to-end AI Development	✓	✗	✗	✗
Direct Monetization Path	✓	✗	✗	✗
Ease of Collaboration	✓	✗	✗	✗
Integrated Governance	✓	✗	✗	✗

# Appendix I: Current State of the Marketplace

As of July 2022, the Tegi marketplace entered closed beta and currently supports a subset of the main ideas presented in this white paper. It is expected that the full set of ideas presented here will take time and iteration to work through into an actual implementation, and we hope that members of the AI community will join in this effort to provide valuable feedback, insight and support in creating a marketplace that serves both the AI community and society at large.

This section gives an overview of what is currently supported in the beta and how users on the beta can expect to use the platform. At present, the following workflow is supported:

1. **Collaboration Creation.** Users may create a new public or private (invite only) collaboration on the platform at any time. Once accepted (currently through a manual review process) the collaboration will be viewable on the marketplace home page (if public) and allow new members to join (again, invite-only if the collaboration is private).
2. **Upload AI.** Once a user has subscribed (joined) a collaboration, they may upload new AI models to the collaboration, which are automatically tested against all resources (i.e. server configurations) that Tegi supports. Once the testing procedure is complete, the model will become available to launch as a new service on any resource that was successfully able to launch the model.
3. **Automated Metrics / Leaderboards.** We are currently testing out automated metrics, currently based on Huggingface datasets and metrics, that enables automated creation of leaderboards, which can be used to automatically select the best available AI model to launch for a new service (e.g. given a resource budget constraint) or to select between AIs.
4. **Launch AI Service.** AI models can be launched in the Tegi cloud and available for use at any time. Tegi supports different resource configurations, with better / faster configurations costing more money.
5. **Use AI.** Once an AI service is launched it can be accessed either on the platform site itself or, more realistically, through the python dev package directly into client applications.
6. **Tegi-supported AI Library.** Similar to most other AI platforms (e.g. AWS, Huggingface), we currently also offer a set of AI collaborations directly supported by Tegi on common tasks such as translation, summarization, or question and answer.

The above features form a complete pipeline which both consumers and developers of AI can look to work together to create custom AI for their use cases. In particular, it is expected that the current implementation is *almost* sufficient to form the following AI collaborations:

1. **Kaggle-style public AI competitions**, with automated scoring and leaderboards, while *allowing direct market access to the winners.*
2. **Fiverr-style private (one-on-one) AI contracts**, with *automated testing and acceptance criteria.*

3. **Cloud-based client applications** that *just get better*, using automated leaderboards to upgrade to new AI on the cloud when they become available.

In comparison to the platform described in this white paper, much of the capabilities, interactions, and roles of key players are not yet supported. In particular, a number of key components are not currently supported in any form:

1. **No git-based control flow.** The final vision of interactions on the marketplace is based around the idea that contracts are mediated through a git-based workflow. While automated testing for feature acceptance is a key part of this workflow (and being concurrently developed), it is currently difficult to have multiple parties work efficiently in the Tegi beta platform, as each component must be deployed as a stand alone-collaboration and direct peer-to-peer payments and rewards are not yet supported.
2. **No third party introspection.** A large part of the motivation for creating the Tegi marketplace was to help streamline and notify the public on how AI is being used, along with mechanisms for automated reports on the impact of AI as well as for manual regulatory introspection.
3. **Many Colony-based infrastructure pieces are missing.** Namely, it is not possible to form partnerships or sub-collaborations (sub-domains), nor is there a reputation based system in place to manage payment to multiple developers in a single collaboration. There are also no multi-party ownership structures that would allow for contributions to a common purse within a budgetary process.

In short, the current platform has very limited support for multi-party collaboration past a single consumer and provider pair, except in the instance that the collaboration is a public leaderboard-style competition. Still, as an active beta the platform does allow for AI development contracts between parties, with automated deployment support such that individual developers may be able to support large clients at scale.

The rest of this section walks through the workflow of a new user arriving on the market website, as well as the steps a consumer would go through to create a new collaboration, have a developer upload novel AI to the collaboration, deploy that AI to the Tegi cloud, and then use that AI locally in a client application through the Tegi dev package.



# Home Page

## Find. Create. Serve.

[Create Collaboration](#) [Download Dev Package](#)

Explore, test and integrate production ready AI already on the marketplace, create a new competition to spur new AI development, or upload your own AI solutions to compete for contracts.

## TEGI Collaborations

AI use case collaborations directly supported by TEGI. Join today to create and upload your own AI, and help develop a people-first AI marketplace!

Uploaded AIs are automatically scored and made available for others to use on the platform, including on open-source applications developed by TEGI. Developers own any AI they upload, and may set their own SaaS fees for clients using their models. Uploaded AIs to competitions are considered private and never made available for download. If you wish to make your AI public, we encourage you to upload it to the HuggingFace Model Hub, which we are dedicated to support (refer to the TEGI dev package documentation for details).



### Translation

Public AI Collaboration

Translation AIs help connect people through our most uniquely human ability—language. This translation collaboration focuses on creating high-quality translation AIs for as many languages as possible, to connect as many people as possible.

8 Models Available

[Join!](#)



### Extractive Summarization

Public AI Collaboration

People are bombarded with more information today than ever before. Summarization AIs help to automatically summarize large volumes of information into smaller, consumable chunks. Extractive Summarization means using only sentences in the source text for summarization.

2 Models Available

[Join!](#)



### Question Answering

Public AI Collaboration

Much of human knowledge is now accessible directly from the Internet or other digital sources. Yet sifting through all this data is time consuming and impractical in many instances—for instance in academic fields where the literature base is too large to practically look through. In these cases AI has the opportunity to help by enabling people to ask simple queries and then...

2 Models Available

[Join!](#)



### Text Generation

Public AI Collaboration

Text generation is a subcategory of NLP (Natural Language Processing). It allows for the automatic generation of natural language.

2 Models Available

[Join!](#)

## AI Library

Common AI use case competitions directly supported by TEGI.

Use the TEGI dev package to incorporate any of the following AIs into your own production application in under five lines of code.

**Translation**  
Translation AIs help connect people through our most uniquely human ability—...  
Public AI Collaboration  
8 Models Available

**Extractive Summarization**  
People are bombarded with more information today than ever before. Summarization...  
Public AI Collaboration  
2 Models Available

**Speech to Text (Real-time)**  
Humans connect to each other most readily through verbal speech. Yet, communicating...  
TEGI Supported AI  
0 Models Available

**Text to Speech**  
Spoken language is the most common form of communication between...  
TEGI Supported AI  
0 Models Available

**Conversational Chatbot**  
Open domain chatbots are used to converse with humans in a variety of contexts. AI...  
TEGI Supported AI  
0 Models Available

**Question Answering**  
Much of human knowledge is now accessible directly from the Internet or other digital...  
Public AI Collaboration  
2 Models Available

**Text Generation**  
Text generation is a subcategory of NLP (Natural Language Processing)...  
Public AI Collaboration  
2 Models Available

## Open Source Apps

Open-source applications, built by TEGI, to get you up and running quickly.



### Real-time Translation

Open Source App

TEGI is dedicated to building tools that bring people together, allow people to better understand each other, and foster technologies that enable people to better empathize with one another. We believe one of the most exciting avenues that AI will enable is direct communication between speakers of different languages. While the technology allowing accurate, real-time translation is still a ways away, we are dedicated to helping build this technology, and have released this end-to-end, real-time translation application to help others seeking to push this technology further.

Whether you are collating datasets for underserved languages, training better models with more compute, creating more intuitive mobile applications, or designing new consumer hardware for...

[Github](#)



### SARS-CoV2 Explorer

Open Source App

The COVID-19 epidemic has negatively impacted virtually every population alive today. Fortunately, the scientific response to the epidemic has greatly sped up our recovery and return to a pre-covid-esque world. While vaccines and other drug therapies have been largely successful, in the past few years tens of thousands of academic publications on COVID-19 have emerged, making it virtually impossible for any single person to read and understand all the knowledge there is on the disease.

Fortunately, Microsoft has released the COVID-19 Open Research Dataset, consisting of tens of thousands of COVID-19 publications, in machine-readable format, for the community to analyze and...

[Github](#)



### Language Practice Partner

Open Source App

The home page is split into three sections,

1. **Collaborations.** All public collaborations are viewable directly through the website. New user collaborations will appear here, and all users may join and participate in these collaborations by navigating to their individual pages from the homepage.
2. **AI Library.** Tegi directly maintains a set of AI collaborations on common AI tasks, such as translation, summarization or question answering. Users may submit new AI to these collaborations, and most of them are expected to generate automated leaderboards using test data curated by Tegi. Users may also use any AI in the library through the Tegi dev package, as with any other collaborative, with the added benefit that Tegi directly supports collaboration-specific *tasks* present in the dev package, generally making their use easier.
3. **Open Source Apps.** As mentioned, we do not currently support git-based workflows within Collaborations directly. We are, however, experimenting with how the marketplace may be used in actual real-world scenarios, and as such are supporting the development of open-source applications that make use of the Tegi marketplace as a collaborative component. Tegi is directly supporting some of these efforts, and is looking for other partners to add to this list as well.

In addition, users may create a new collaboration or download the dev package through the buttons at the top of the page.

Users may navigate to any collaboration page by selecting the appropriate collaboration, or navigate to their own dashboard (using the menu in the top right) to manage their account information, launch new AI services or check their billing and payment.

## Creating a New Collaboration

Any user is allowed to create a new collaboration. To do so they may click on the *Create Collaboration* button at the top of the homepage. From there they are taken to a new page where they name and declare the type of the collaboration, which will determine which options and settings are available.

## Create Collaboration

Question Answering  
<https://marketplace.tegi.ai/collaborations/jeremy-wurbs-question-answering>

**Collaboration Type**

Public AI Collaboration

Detailed Information

**Create Collaboration**

I have read and accept the Terms of Service

Once the collaboration is created, it will be visible to the creator (e.g. if they log out and log back into their account), and accessible again through their own dashboard, but will not be available or visible to other users on the platform until the collaboration information page is filled in and approved by a Tegi administrator. In particular, every collaboration must currently define an input-output relationship for any AI submitted to the collaboration.

## Collaboration Details

Question Answering Public AI Collaboration  
<https://marketplace.tegi.ai/collaborations/jerry-writes-question-answering>

### Privacy Settings

**Search**

If Search is set to Public, the collaboration will be visible and searchable from the Tegi platform home page. If Search is set to Private, users will only be able to find, view and sign up for the collaboration via a private link provided by the collaboration owner.

**Providers**

If Providers is set to Public, anyone will be allowed to join the collaboration and submit AI solutions. If Providers is set to Private, users will have to request approval from the collaboration owner before they are allowed to join and provide new solutions.

**Consumers**

If Consumers is set to Public, anyone will be allowed to join the collaboration and use AI solutions. If Consumers is set to Private, only the collaboration owner will be able to use any submitted AI solutions.

### Summary Description

Provide a short description of the use case and expected collaboration output.

### Acknowledgements

Provide a brief introduction to the host and any helping organizations, including funding agencies providing support.

### Broader Impact Statement

Include a statement of the potential broader impact of this collaboration, including information on where and how resulting AI will be used.

## Pipeline

A collaboration is defined by a Pipeline. A pipeline is a single input-output relationship, where both the input and output are dictionaries with fixed keys and known data types for each value. Any AI or service submitted to this collaboration must support the pipeline defined here.

### Create Pipeline

**Inputs** **Outputs**

Pipelines require an input and output.

## Data & Evaluation

All data and code is optional and only needed as makes sense for a particular collaboration. Training data will be downloadable by any user that subscribes to this collaboration. If the Providers privacy setting is set to Private, users will have to be approved before they can download the data. Any provided code is made available according to the same training data privacy rules. Code is optional, but must be provided in two cases: (1) when the data is a proprietary format and users cannot reasonably be expected to know how to load it properly, the collaboration host must provide data loader samples / documentation, and (2) when the collaboration includes standardized metrics, the code to compute the metric must be provided by the collaboration host (if the code is not otherwise publicly available). Evaluation data is only used for testing solutions and is never made publicly available.

Upload Code

Upload Data

### Automated Evaluation

Tegi currently supports automated testing for all HuggingFace datasets as well as select HuggingFace metrics. Any number of dataset-metric pairs may be added, with all submitted AIs automatically tested on each pair. Automated tests will create a leaderboard for all submitted AIs and services, which can be used to select the best available AI automatically at runtime.

Refer to <https://huggingface.co/datasets> for available datasets.

**Dataset**  **Split**

**Subset**  **Metric**

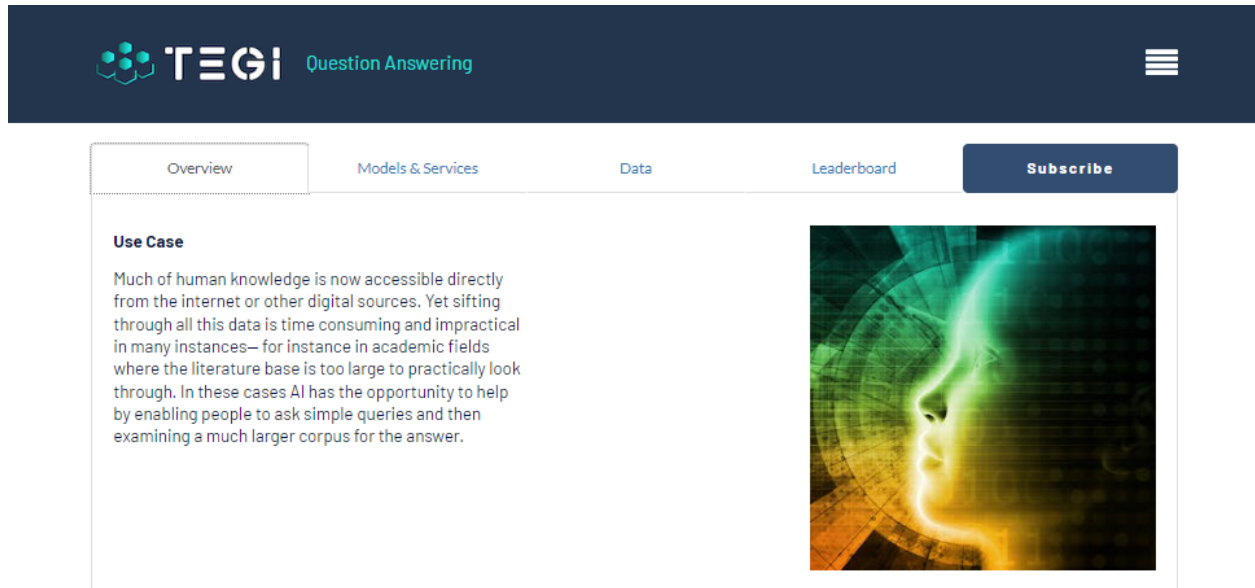
### Detailed Data & Evaluation Information

**Data Details**

Describe the data format, along with any included code to load, pre-process, transform, or analyze the data.

After all the collaboration details are complete, the user may hit *Submit Collaboration*, at which point it will be marked for manual review. Once a Tegi administrator approves the collaboration it will be accessible to all users on the marketplace, either as a public or private collaboration, as defined by the creator.

## Subscribing to a Collaboration



The screenshot shows the top navigation bar of the TEGI Question Answering platform. The bar is dark blue with the TEGI logo and 'Question Answering' text on the left, and a hamburger menu icon on the right. Below the navigation bar is a horizontal menu with five items: 'Overview', 'Models & Services', 'Data', 'Leaderboard', and 'Subscribe'. The 'Subscribe' button is highlighted in a darker blue. The main content area is divided into two columns. The left column has a section titled 'Use Case' with the following text: 'Much of human knowledge is now accessible directly from the internet or other digital sources. Yet sifting through all this data is time consuming and impractical in many instances— for instance in academic fields where the literature base is too large to practically look through. In these cases AI has the opportunity to help by enabling people to ask simple queries and then examining a much larger corpus for the answer.' The right column contains a square image of a glowing human head profile in profile, with a bright yellow and green light emanating from the forehead area, set against a dark background with faint grid lines.

Users must have access to view and subscribe to a collaboration. Public collaborations are viewable to all users, while private collaborations require the creator to add participants directly, at which point they will be able to view and subscribe to the collaboration as any other.

The user may subscribe to a collaboration by navigating to the collaboration's page and clicking on the *Subscribe* button in the upper right and accepting the collaboration's Terms and Agreements.

# Uploading AI to a Collaboration

The screenshot shows the TEGI Question Answering interface. At the top, there is a dark blue header with the TEGI logo and the text "Question Answering". Below the header, there is a navigation bar with tabs for "Overview", "Models & Services", "Data", "My Page", and "Leaderboard". The "My Page" tab is currently selected. The main content area displays the profile of "Jeremy Wurbs".

**Jeremy Wurbs**  
Joined: Apr 29, 2022 1:55 pm  
Profile: <https://marketplace.tegi.ai/users/jeremy-wurbs>  
About: Tegi Founder and CEO

**Upload**

roberta-base-squad2 Confirm Name

roberta-base-... Progress bar

Once a user has subscribed to a collaboration, a new *My Page* tab will appear, which will allow them to upload new AI to the collaboration. In general, it is expected that the creator of the collaboration will provide test data and code, as required for a developer to know that their AI model has been implemented correctly. In any case, once the AI model is uploaded, Tegi will automatically test which resources (i.e. set of supported CPUs/GPUs) that model may be run on and, if possible, run any automated tests set up by the collaboration owner to give the new model a benchmark score.

# Launching a New AI Service (Endpoint)

The screenshot shows the Tegi Question Answering dashboard. The top navigation bar includes the Tegi logo, the text 'Question Answering', and a hamburger menu icon. Below the navigation bar are five tabs: 'Overview', 'Models & Services', 'Data', 'My Page', and 'Leaderboard'. The 'Models & Services' tab is active. The main content area is divided into three sections: 'Search & Delete', 'Launch New Service', and 'Test & Compare'. The 'Search & Delete' section has a message: 'You have not launched a service yet, but when you do you may find, view and delete them here.' The 'Launch New Service' section has a 'Model' section with a dropdown menu showing 'roberta-base-squad2' selected and 'bert-large-uncased-whole-word-masking-finetuned-squad' as an option. Below the model dropdown is a 'Select Resource' dropdown and a checkbox for 'Resource Price List'. The 'Deploy' section has input fields for 'Idle Time Out' (1800) and 'Fixed Time Out' (N/A), each with a checkbox for 'Details'. There is also a 'Service Name' input field and a 'Deploy' button. The 'Test & Compare' section has a message: 'There are currently no other active services for this collaboration. Launch a new one to test it here.' and an 'Add' button.

In order to use AI on the Tegi platform, it must be loaded into a Tegi *Service* (i.e. an endpoint). Once an uploaded model passes all automated testing (or, at least, one resource check), that model will be available to launch as a new service in one of two (otherwise identical) ways:

1. **Web platform.** There are two places on the platform a user can launch a new service:
  - a. *Models & Services* tab on the collaboration page, or
  - b. *Launch New Service* section within the user's dashboard.
2. **Dev Package.** Users may also launch new services directly through the dev package (refer to the associated documentation).

*Note:* Users will only be able to launch AI models that they have the permission rights to do so. Generally, that means they may launch any Tegi-supplied model, any AI they have uploaded to their own account, any AI from public collaborations, and any AI from (private) collaborations they created.

### Search & Delete

You have not launched a service yet, but when you do you may find, view and delete them here.

### Launch New Service

#### Model

roberta-base-squad2

#### Resource

Tier 1  Resource Price List

Tier	Information	Price (USD)	Price Unit
Tier 0	CPU Only, 8GB Memory	0,10	PerHour
Tier 1	CPU Only, 16 GiB Memory	0,50	PerHour
Tier 2	GPU, 16GiB Memory	1,00	PerHour
Tier 3	4x GPU, 192GiB Memory	8,00	PerHour

#### Deploy

Idle Time Out:   Idle Time Details

Fixed Time Out:   Fixed Time Out Details

Roberta Q&A

Launching services is not free, with the cost dependent on the resource tier selected. Idle and fixed time outs may be supplied to assure that the user does not accidentally keep an idle endpoint up indefinitely (currently idle time outs are required to be set during the beta).



Overview Models & Services Data My Page Leaderboard

### Search & Delete

Search for a model InService From: 2022/01/01 To: 2023/01/06

Service	Collaboration	Model	Service ID	Status	Creation Date	Deletion Date
Roberta Q&A	Question Answering	roberta-base-squad2	1672922347474x361060617027584000	InService	2023/01/05, 22:39:22	Delete

### Launch New Service

**Model**  
Select Model

**Resource**  
Select Resource  Resource Price List

**Deploy**  
Idle Time Out: 1800  Idle Time Details  
Fixed Time Out: N/A  Fixed Time Out Details  
Service Name

### Test & Compare

Select Service

Roberta Q&A

Once a service is marked as *InService*, it will be available for real-time inference on demand, either through the website (via the *Test & Compare* section) or, more likely, through the dev package. Note that there are at least a few benefits for users to test services directly through the webpage, as there all *InService* services may be tested simultaneously, allowing the user to play around with and compare different AI (or the same AI on different resources) in a simple, timed GUI, while also providing the user the raw input and output sequences (which may be required for correct use in the dev package).

TEGI Question Answering

Overview Models & Services Data My Page

**Jeremy Wurbs**

- Tegi
- Dashboard
- Collaborations
- Log out

Also note, all of the above (launching, monitoring and deleting services) may also be accomplished through the user's *Dashboard*, accessible through the menu in the upper right of the page header.

## Using AI Services

Access to using AI on the Tegi platform is expected to be conducted through the Tegi (python) dev package. Currently the dev package is only available to beta users, but it is expected to be open-sourced and freely available through PyPI in the future.

The Tegi dev package may be installed directly through pip with

```
pip install tegi
```

Then, in order to use AI on the platform, users may do something similar to the following (please refer to the package documentation for full details on usage),

```
user_collaboration.py

from tegi import User, Collaboration

user = User(email=config.email, password=config.password)
collab = Collaboration('jeremy-wurbs-question-answering', user.api_token)
```

The above will log in a user (receiving a new API token, which can then be used to access and create new collaborations and services), as well as create a new instance of a *Collaboration* object. Note that all collaborations on the platform have a unique name, generally of the form *user-name-collaboration-name*, which can be used to identify collaborations in code.

By default, Collaboration objects will use the “best” pre-existing AI service available to the user. If no service is available, a new one may be launched with:

```
user_collaboration.py

collab.deploy(model=model_id, resource=resource_id, user_api_token=user.api_token)
```

where *model\_id* and *resource\_id* may be found on the Tegi website or queried directly through,

```
user_collaboration.py

models = collab.available_models()
for model in models:
    print(f'model {model.id}: resources {model.available_resources()}')
```

For common use cases, Tegi provides and supports ready-made pipelines. In these cases a *Pipeline* object is provided for the use case, which is itself a collaboration which can be found on the Tegi marketplace, like any other.

```
extractive_summarization.py

import hydra

from tegi import User, ExtractiveSummarizationPipeline

text = "It is not the critic who counts; not the man who points out how the strong man
stumbles, or where the doer of deeds could have done them better. The credit belongs to the
man who is actually in the arena, whose face is marred by dust and sweat and blood; who
strives valiantly; who errs, who comes short again and again, because there is no effort
without error and shortcoming; but who does actually strive to do the deeds; who knows great
enthusiasms, the great devotions; who spends himself in a worthy cause; who at the best knows
in the end the triumph of high achievement, and who at the worst, if he fails, at least fails
while daring greatly, so that his place shall never be with those cold and timid souls who
neither know victory nor defeat."

@hydra.main(config_path='../tegi/configs', config_name='user')
def run_model(config):
    user = User(email=config.email, password=config.password)
    model = ExtractiveSummarizationPipeline(user_api_token=user.api_token)

    summary = model(text)

    print('Excerpt:\n' + text + '\n')
    print('Summary:\n' + summary)

if __name__ == '__main__':
    run_model()
```