General information:

- Name of project
  - Is the project's name new or an existing name? **New**
  - **OCPP Cloud Connector**
- Project description (what it does, why it is valuable, origin and history)
  - **Origin: My company, ChargeNet, has been looking for a cloud-based publish/subscribe module that takes in OCPP data delivered from the charger and delivers it to a data streaming platform such as Kinesis, Kafka, Cloud Dataflow, or Confluent. We initially bought a license to software developed by a 3rd party, but found it did not have key functionality we want in a production environment.**

    We searched the open-source projects and found nothing that fit our need of a cloud-based publish/subscribe OCPP connector.

    This OCPP Cloud Connector is not a core value-add (ie, it is not IP) of ChargeNet, and given multiple business types could benefit from a pub/sub connection to the chargers, we would like to make the software open-source.

- Project lead
  - **ChargeNet? Not sure how this works**
- Project financial sponsor organization(s)
  - **N/A**
- Names of other key contributing individuals and organizations
  - **ChargeNet: Rebecca Wolkoff, Morgan Murphy, Leah Erb**
  - **Volta Foundation: Gabe Hege**
  - **Jeremy Huether**
- Proposed Technical Steering Meeting (TSC) members
  - **?**
- Existing community links:
  - repository hosting **(request GitHub)**
  - project website and docs
  - mailing lists, slack, irc
  - social media accounts
  - **None exist at this time**
- Project security plan
  - **We will adhere to typical and recommended practices, including:**
    - **2-step authentication**
    - **Never storing credentials on github**
    - **Doing code-reviews before any commit to a master branch**
  - **We would be interested in hearing other recommendations from LF Energy and the leadership team**
- Link to code base
  - **?**

Open source status:

- Please describe the project's license.
  - **The license is TBD but will likely be https://choosealicense.com/licenses/apache-2.0/**
- Is the project's code available now? **Has not been written.**
  - Is the code publicly posted? On GitHub or elsewhere? **No.**
- Does this project have ongoing public (or private) technical meetings?
  - **It absolutely will.**
- Do this project's community venues have a code of conduct? If so, what is it?
  - **Respect and inclusion are important values. We are willing to adopt a code of conduct that encourages these two.**
- Describe the project's leadership team and decision-making process.
  - **Open to discussion. We do sprint planning every two weeks and create stories at this time. Stories may also be created ahead of time and sprint planning is a time to bring up and discuss what the priorities are.**
- Does this project have public governance (more than just one organization)?
  - **We are very open to working with other organizations**
- Does this project have a development schedule and/or release schedule?
  - **MVP by April, likely**
- Does this project have dependencies on other open source projects? Which ones?
  - **?**
- Describe the project's documentation.
  - **Possibly in an external Slab as well as Jira for story tracking. We can have:**
    - **An area for meeting minutes**
    - **Links to Jira**
- Describe any trademarks associated with the project.
  - **None**

Project status:

- Do you have a project roadmap? please attach [Are this project's roadmap and meeting minutes public posted?]
  - **We will create a roadmap in Jira in the first meeting**
  - **The OCPP Cloud Connector should be a cloud-based publish/subscribe module that takes in OCPP data delivered from the charger and delivers it to a Kinesis stream. Along with having the ability to deliver data from the charger, it should also be able to send instructions to a charger. We currently have the following requirements for the OCPP connector module:**
    - **Ability to send complex charge instructions to a charger**
    - **Deployed as a lambda or in a docker container to ECS Fargate**
    - **Ideally does not store any data**
    - **Reads/writes directly to and from Kinesis**

- **Ideally written in Go, Python, or Typescript (with Go being the most preferred, but open to discussion/feedback)**
- **Should support WS and WSS**
- **Support OCPP 1.6J and 2.0 in ways that are helpful to optimization. Specific support needed:**
  - **Smart charging (send charge profiles)**
  - **Remote start and remote stop**
  - **Receive events when a car (local start and stop)…**
    - **Is plugged in**
    - **Unplugged**
    - **Charger changes status**
  - **Configuration over OCPP (can change a configuration or get a configuration)**
  - **Support command to disable and enable a plug**
  - **OCPP 2.0: want the features to get more info (or any info available) about the car specifically**
  - **Ability to lock or unlock a plug**
  - **Authorized-specific commands such as payment authorization and authorization to start a charge**
- Does this project have a legal entity and/or registered trademarks? **No**
- Has this project been announced or promoted in any press? **Not yet**
- Does this project compete with other open source projects or commercial products? **From what we've researched, this replaces software that other companies may offer for $30k-$50. However, I have never seen these publicly offered.**

Project value:

- Why would this project be a good candidate for inclusion in LF Energy?
  - **This project addresses a need in the electric vehicle charging community. Any potential company, project, or person that would like to communicate with or send commands to a charger would be able to use this repository to do so. It would allow small electric vehicle charging businesses to more easily compete with larger corporations, and it would allow students and innovators to easily interact with chargers.**
- Provide a statement on alignment with the mission in the LF Energy charter.
  - **This directly relates to the distribution and delivery of energy.**
- What specific need does this project address?
  - **We searched the community for a publish/subscribe OCPP connector repository and did not find one. This addresses the need in the community for an event-driven OCPP connection.**
- Describe how this project impacts the energy industry.
  - **Smaller businesses wanting to interact with an OCPP charger could compete more easily because this repository existed.**
  - **Students could easily do projects using this repository to control and receive messages from a charger.**
- Describe how this project intersects with other LF Energy projects.

- - I understand there may be other projects directly developing OCPP code for the chargers. This is an interface to be run in the cloud between a charger and any communication/control software.
  - Who are the potential benefactors of this project?
    - ChargeNet and small businesses entering the EV charging industry have the most to benefit. I imagine large businesses that may have developed an interface early on may be interested in using this one as it may provide better maintenance as the transition to OCPP 2.0 happens.
    - Students / classrooms that have an OCPP-capable charger would also benefit as this software would allow for students/teachers to easily interact with a charger.
  - What other organizations in the world should be interested in this project?
    - Possibly big players: Engie, Stem, Enel X, ChargePoint, EVGo, EVConnect
    - Possibly charger manufacturers interested in entering the control space
    - Smaller organizations: Enode,

Project needs:

- How would this project benefit from inclusion in LF Energy?
  - We have never organized an open-source codebase before. We could absolutely use help on organization, leadership, and standardization of the code-base. We also think LF Energy is fairly well known (or becoming more well known) in the industry.
- Please describe any infrastructure needs or requests (e.g., web hosting).
  - GitHub Pages
- Plan for achieving next maturity level (Incubation -> Early Adoption -> Graduated).
  - Incubation:
    - We hope to have a fairly well-working repository by the end of April 2022
    - After initial organization and set-up of the repositories, ChargeNet and leadership team will story out what needs to be accomplished, including tests that need to be written
    - We will host sprint planning every two weeks and stories as well as acceptance tests will be outlined in Jira
    - We will advertise in small networks to gain some interested experienced developers, for example to the LACI and Plug & Play incubators
  - Early Adoption
    - Once we have a fairly solid code-base, we will spread the word on social media widely to gain more contributors and suggestions
  - Graduated
    - I think there will be continuous work on this codebase for years to come as further EV connectivity is built out and OCPP standards are updated

- **As of now, we will consider this codebase graduated when it supports full OCPP 2.0 functionality, which we imagine will likely take upwards of a year**