Team Apollo-Report Out Concept Generation
Elisha Aguilera, Setenay Gunner, Denish Oleke

Client Needs:

The project sponsor and potential client specified an array of collected needs including:

A system of managing project dependencies (1) which can be internally deployed by a client without depending on external hosting (2), which involving the exchange of version-controlled resources between authenticated clients and the hosting dependency registry (3). This entails that dependencies can be registered from an authenticated client or pulled from a third-party packages which can be verified before marked as valid dependencies for projects.

Product Review:

Based on video conference exchanges over four meetings so far, we have received some technical recommendations from the sponsor indicating that our intentions to use Python 3 with the ArgParse library for the command line client implementation is approved, but it was recommended to use the Go Language standard library implementation for an HTTP server to provide endpoints and dependency building for handling the server component of the project.

Our own research indicates that despite our team's lack of experience with Go and its standard library, it possesses a syntax very similar to C-inspired languages and should be relatively easy to learn and incorporate into the project, which may be beneficial as its design was optimized for distributed computing.

Architecture Review:

Given the networked nature of this project's deliverables, a standard multilayer architecture was utilized for our project architecture design, shown graphically below in *Figure 1*:

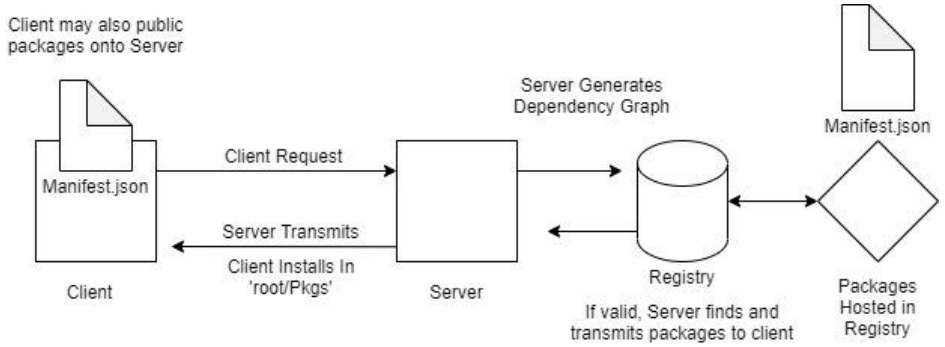

Figure 1 – Multilayer Architecture prototype layout

All client components will be implemented using Python 3, with the business layer consisting of the server, implemented in Go Lang, itself interacting with a registry database in the information layer using an undetermined existing database system, as the team will have to evaluate the benefits and drawbacks of a relation or document-based database.

Further Research Needed conclusions:

As mentioned in the Architecture Review section, what technology will be used for the registry database is yet to be determined; as the project is developed, certain issues will arise that may entitle using different technologies or architecture modifications to compensate.