Pivotal Interface for Kubernetes

Vision Statement

<table>
<thead>
<tr>
<th>Position</th>
<th>Name</th>
<th>Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team Lead</td>
<td>Jesmar Castillo</td>
<td><a href="mailto:jesmar@ucsb.edu">jesmar@ucsb.edu</a></td>
</tr>
<tr>
<td>Team Scribe</td>
<td>Marco Chavez</td>
<td><a href="mailto:mchavez00@ucsb.edu">mchavez00@ucsb.edu</a></td>
</tr>
<tr>
<td>Developer</td>
<td>Durva Kapadne</td>
<td><a href="mailto:durva@ucsb.edu">durva@ucsb.edu</a></td>
</tr>
<tr>
<td>Developer</td>
<td>Jack Liu</td>
<td><a href="mailto:jackliu@ucsb.edu">jackliu@ucsb.edu</a></td>
</tr>
<tr>
<td>Developer</td>
<td>Kindy Tan</td>
<td><a href="mailto:ktan@ucsb.edu">ktan@ucsb.edu</a></td>
</tr>
</tbody>
</table>

Product Name: **kubernetes konekt**  Team Name: **The Goodfellas**

Background

Kubernetes is a platform that is made to revolutionize the way that container based applications are released and tested. Kubernetes makes use of Docker containers that package and run an application along with all its dependencies in an isolated environment, eliminating errors experienced from running an application on different machines. Unlike a regular virtual machine, Docker containers do not need a guest operating system, making deployment lightweight.

Kubernetes builds on top of this by automating the deployment, scaling, and management of containerized applications within a cluster of computers. Through a master server, a cluster and its nodes can be managed, distributing containers among the nodes.

Problem Statement

The lack of connectivity between users with containers and users with clusters acts as a barrier for deployment. Currently there is not an interface that streamlines connecting these two groups.
Problem Outcome

The Minimum Viable Product (MVP) will be a desktop interface that will connect users all over the world and allow them to distribute or recieve containers. As of now, the workflow is very simple, and we want users to have the available functionality to customize how they organize their software and control who they distribute it to. This will allow for better user experience and a more flexible application. The solution may expand into a web or mobile application, allowing users to schedule their containers on the go.

Milestones

- Research Kubernetes, Pivotal tools, Docker, and understand their usage and applications.
- Begin a frontend and backend user interface, with divided work between the whole team.
  - Implement a workflow standard for the whole team to work on.
- Desktop application to connect users with containers to users with clusters.
  - Implement backend skeleton to track location of the kubernetes (k8s).
  - Create UI skeleton to display data and manage containers in an orderly fashion.
- Web UI for cross-platform capabilities.
- Mobile application to easily manage user containers on the go.

How we plan to articulate and design a solution

- Daily meetings with team and weekly meetings with Pivotal mentors to update the progress of the project and discuss further plans to move project forward. This will generally consist of:
  - Assigning tasks (during sprints)
  - Update on overall project as well as individual tasks
  - Discussing strategies on team working together to improve productivity
  - Receiving feedback from company employees on the product and changing it along the way

- Docker
- Google Computer Platform
- Pivotal Tools
  - Concourse CI
  - Pivotal Container Service
  - Pivotal Tracker