Towards Enhancing Usability and Modularity of the PRAGMA Cloud Scheduler

Nannapas Banluesombatkul[1], Prapansak Kaewlamul[1],
Shava Smallen[2], Nadya Williams[2], Prapaporn Rattanatamrong[1]
[1] Thammasat University, Thailand. [2] University of California, San Diego.

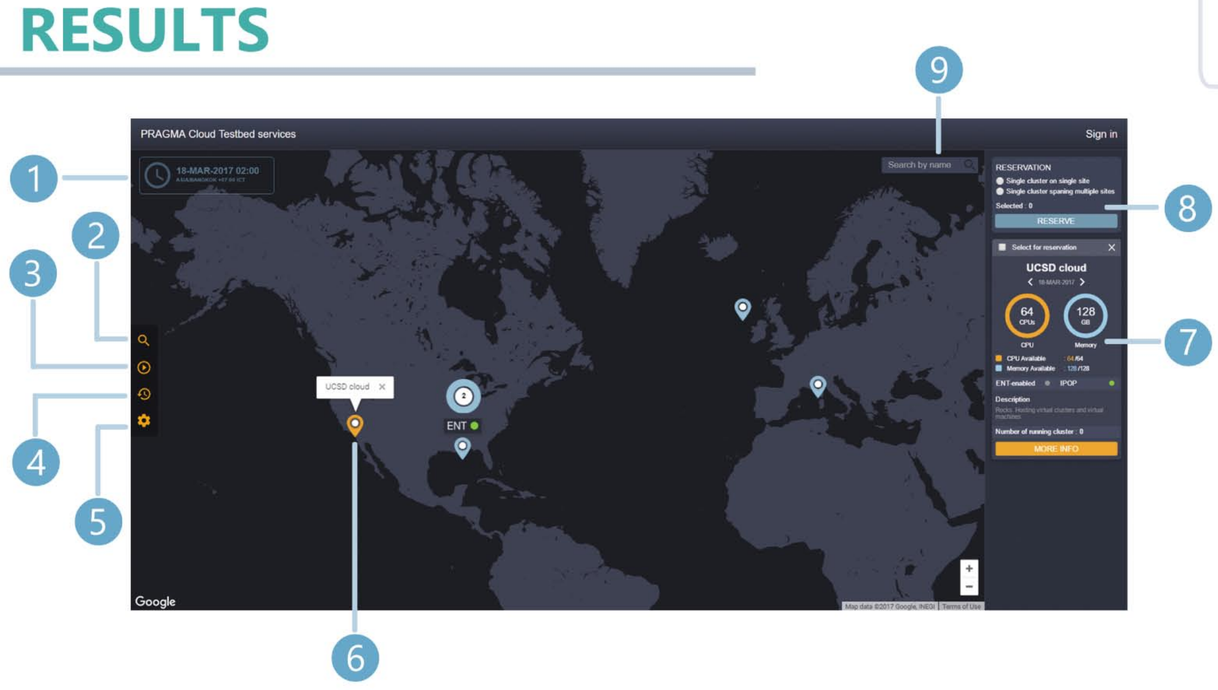
MOTIVATION

- The PRAGMA Cloud Scheduler provides researchers the capability for reserving shared computing resources which are distributed in many sites.
- There is the limitation of the PRAGMA Cloud Scheduler in supporting broader group of users, including those who might not have intensive computing background.
- The Resource Working Group also highlighted the possibility of ongoing PRAGMA/CENTRA projects bringing in new types of resources to share such as sensors or data storages.

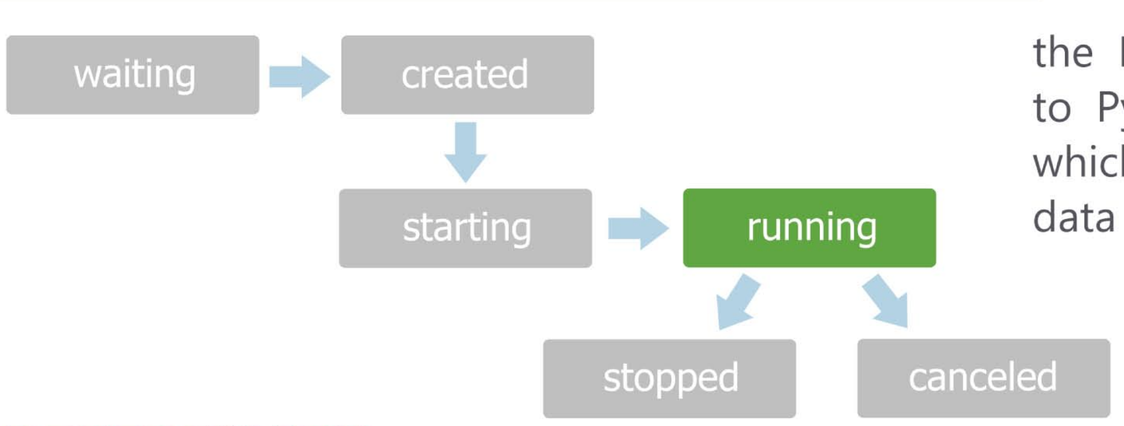
OBJECTIVE

We propose the new design and implementation of the PRAGMA Cloud Scheduler with enhanced usability and modularity.

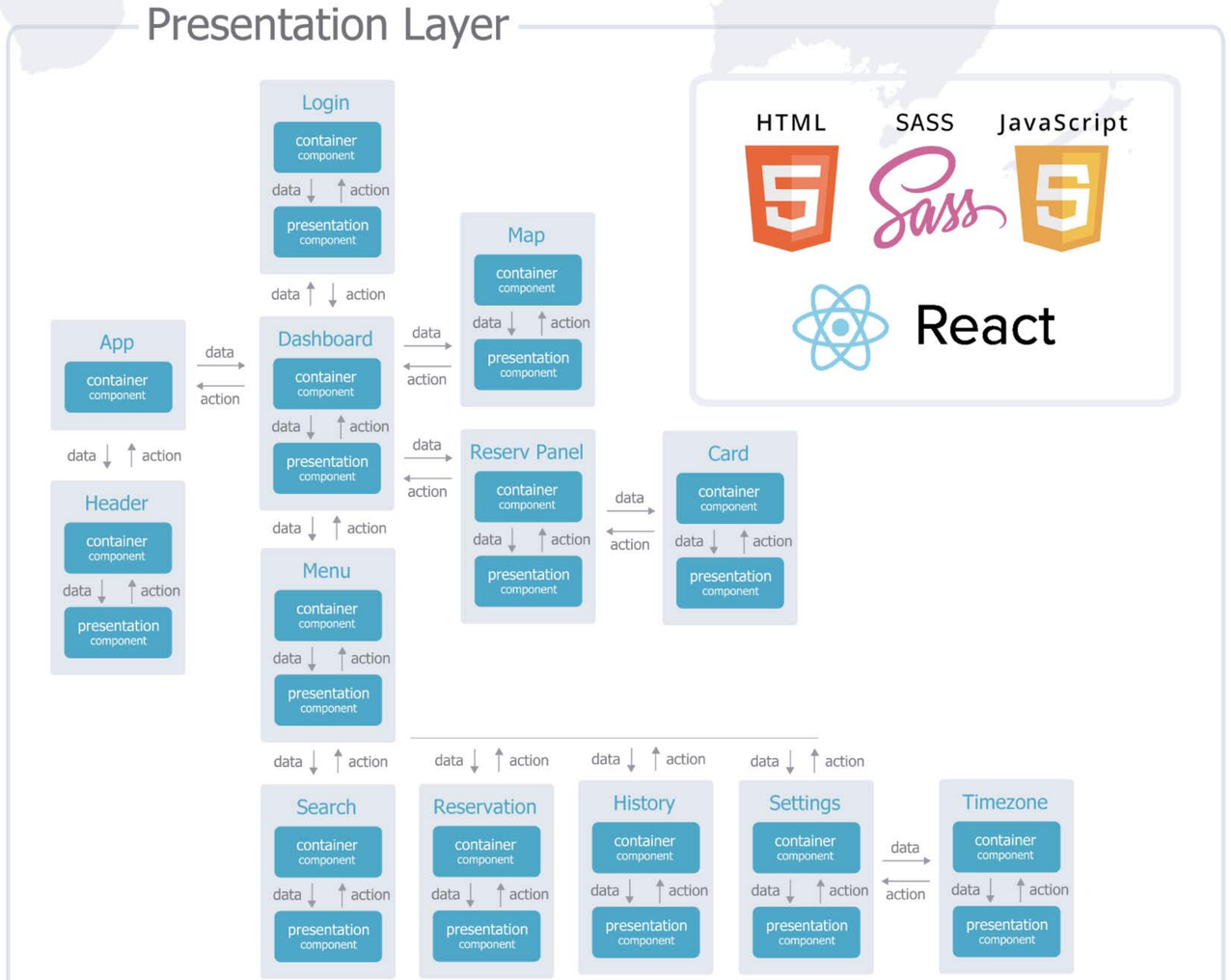
RESULTS

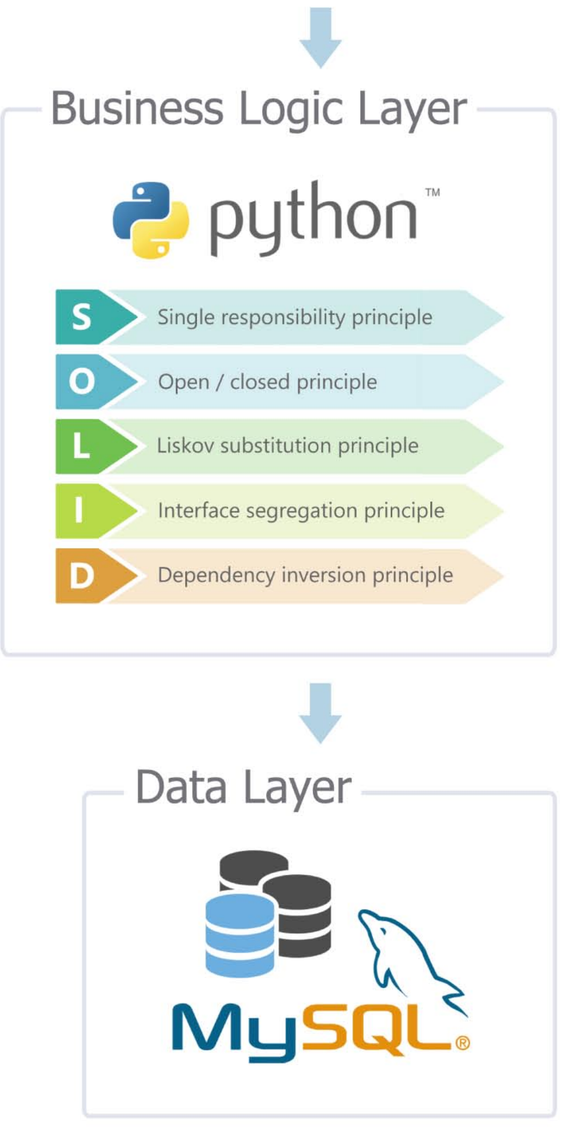


STATES OF A CLOUD RESERVATION



SYSTEM ARCHITECTURE





Function of PRAGMA Cloud Scheduler

1. Display a user's timezone
2. Search sites by any criterias
3. View running reservations
4. View reservations history
5. Settings
6. See sites' locations on map
7. Get sites' descriptions
8. Create a reservation
9. Search sites by site's name

When users interact with the UI, the Presentation layer sends the request to Python scripts in Business Logic Layer which determines logic, changes or services data and returns to Presentation Layer.

REFERENCES

Shava Smallen, Nadya Williams, Philip Papadopoulos. (n.d.). **Lightweight Scheduling for the PRAGMA Cloud Testbed**. San Diego: University of California.

Lormongkol, V. (2016, May 9). **SOLID, The OOD Principle.** Retrieved December 15, 2016, from 2bedev: https://2bedev.com/solid-the-ood-principle/

CONCLUSION

This web application is ready to use with all basic functionalities to meet the requirements, such as creating a reservation of resources, getting a list of service sites and viewing the reservation status. Moreover, We plan to examine the user friendliness of the system by using a usability test with real users and be open for further expansion without excessive code modification.