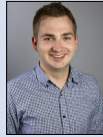




Dijan Helbling



Roman Kramer

| | |
|--------------|--|
| Students | Dijan Helbling, Roman Kramer |
| Examiner | Dr. Thomas Bocek |
| Subject Area | Internet Technologies and Applications |

Shared Processing Unit

Reshaping Web Monetization

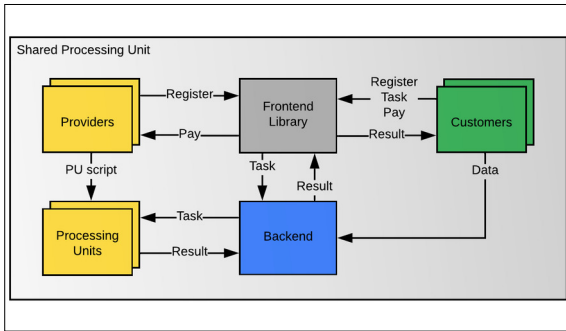


Figure 1 White Box
Own presentation

Introduction: Web Tracking and Personalization plays an important factor in today's marketing solutions. The consumer is more likely to purchase something they are interested in, which in turn leads to higher revenue. A lot of consumers do not like this kind of Web Tracking, which can be observed from an increase of Ad Block users. However, the lack of ads creates a gap in revenue on the internet, which can affect websites that are providing free content.

Shared Processing Unit is a concept to reshape the monetizing system on the internet. SPU provides a form of revenue to website providers without intruding on the website visitor's privacy. SPU takes advantage of two factors. Firstly, the unused processing power of internet-capable devices and secondly, the need for more processing power for data analytics. SPU can combine both factors through a shared network of Processing Units.

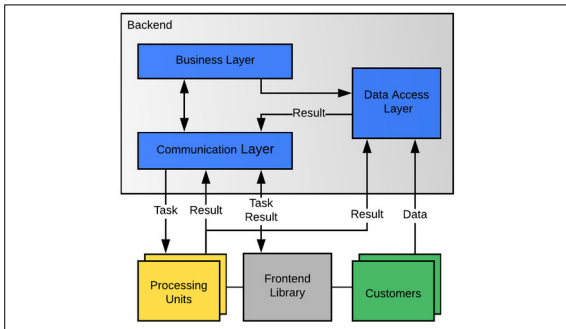


Figure 2 Backend View
Own presentation

Objective: SPU introduces a revenue stream on the internet without intruding the visitor's privacy. SPU does not require any form of web tracking since the quality of a processing unit is independent of a website visitor's identity. The quality of a processing unit lies in the amount of processing power provided. Providers can improve the quality of their website by removing ads and giving a non-distracting user experience. To ensure said user experience, excessive power consumption and data usage must be avoided. Entities that use the SPU network to run algorithms benefit from cheap calculation power and faster computation time. There is no need for prior knowledge about algorithms, all the customer needs is a data set.

Conclusion: The main components, such as the processing unit, backend, algorithms and frontend, were all implemented while only considering the main use cases. The architecture was revised multiple times during development due to requirement changes. However, the final architecture design was decided in the eighth week, which allowed to fully implement the prototype. To bring a minimum viable product to the market all the functional requirements as listed in Figure 3 needs to be implemented. Future work will investigate if the concept is viable for actual business use cases.

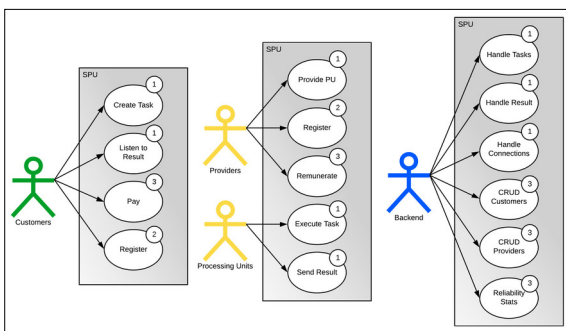


Figure 3 Use Case Diagram
Own presentation