

### *Abstract*

*Monolithic architecture designing application systems consists of only one large part and runs on one computational instance [1]. This causes, the process of improving or repairing the application system involving the whole system. Heartenly.com as a matchmaking application from Indonesia, still uses monolithic architecture Heartenly.com, requires an architecture that can improve or repair the system independently without having to turn off all services. For this reason Heartenly.com changes its architecture to Microservice. The process of change involves a decomposition process to break down the services that are on Heartenly.com. The decomposition method uses the principle of Domain-driven Design (DDD) combined with the Representational State Transfer (REST) architecture to optimize communication lines because Heartenly.com is web-based. The results of decomposition are tested by the method of the Single Responsibility Principle (SRP) and the Common Closure Principle (SRP) adaptation of the principle of Object Oriented Programming (OOP). In this research, the results of the analysis prove that microservice can do improvement and deployment independently*

*Keywords: software architecture, microservice, domain-driven design, REST*