



Katedra Inżynierii Oprogramowania  
kierunek studiów: Informatyka  
specjalność: Inżynieria Oprogramowania

Praca dyplomowa inżynierska

**IMPLEMENTACJA APLIKACJI DO WSPÓŁTWORZENIA  
NOTATEK Z WYKORZYSTANIEM TECHNOLOGII PEER TO  
PEER**

**IMPLEMENTATION OF A COLLABORATIVE NOTE-TAKING APPLICATION  
USING PEER-TO-PEER TECHNOLOGY**

**Oskar Marcin Chybowski**

nr albumu: **54941**

Opiekun:

**dr inż. Mirosław Mościcki**

Katedra Inżynierii Oprogramowania

Szczecin, 2026



## **Contents**

Zakres teoretyczny .....	4
Systemy współdzielenia dokumentów w czasie rzeczywistym .....	4
Ewentualna zbieżność .....	4
Silna zbieżność .....	4

## **Zakres teoretyczny**

### **Systemy współdzielenia dokumentów w czasie rzeczywistym**

#### **Ewentualna zbieżność**

Budując rozwiązania związane z równoczesnym tworzeniem i modyfikacją jakichkolwiek danych przez więcej niż jednego użytkownika, musimy rozważyć mechanizmy zapewniające conajmniej ewentualną zbieżność (eventual consistency), która gwarantuje, że każda kopia danych w naszym systemie będzie identyczna i zawierała najnowszą zmianę u każdego klienta po otrzymaniu wszystkich zdarzeń modyfikujących dane.

Popularną strategią w takim podejściu jest Last-Write\_wins (LWW). To podejście, gdzie każde otrzymane zdarzenia są ułożone w pewnej kolejności, zazwyczaj opartej o czas. Rozstrzyga ona konflikty poprzez nanoszenie tylko tej zmiany, która jest uznawana jako ostatnia w kolejności zbioru konfliktujących operacji. Ustalanie kolejności nie jest jasno tutaj zdefiniowane. W systemach baz danych takich jak Cassandra oraz SQL Server P2P każdy zapis otrzymuje własny znacznik czasowy, na podstawie którego wybierany jest najmłodszy wpis i nim nadpisywane są zmiany w źródle danych. Zmiany ze starszymi znacznikami są porzucane. Zauważalną wadą LWW jest wysokie ryzyko utraty danych w czasie nanoszenia zmian, ponieważ wszystkie starsze zmiany nie są brane pod uwagę.

doi:10.1145/1435417.1435432 <https://cassandra.apache.org/doc/latest/cassandra/architecture/dynamo.html#data-versioning> <https://learn.microsoft.com/en-us/sql/relational-databases/replication/transactional/peer-to-peer-transactional-replication?view=sql-server-ver17> Designing data intensive applications - ISBN 1491903112, 9781491903117

#### **Silna zbieżność**

W przypadku systemów oferujących współbieżną modyfikację tekstu, gdzie możliwość wystąpienia konfliktów zwiększa się znacząco - przez większą ilość klientów oraz łatwy dostęp do obszarów aktualnie edytowanych przez innych użytkowników - powinniśmy rozważyć mechanizmy silnej zbieżności.