

The QuoteServices App Case Study: Implementing Zippy Microservice (Part 1)

Douglas C. Schmidt

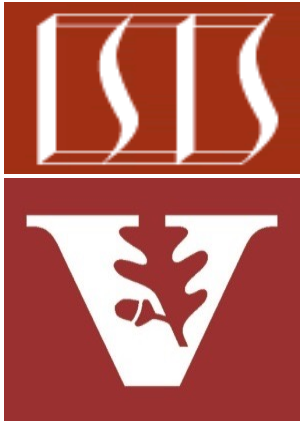
d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

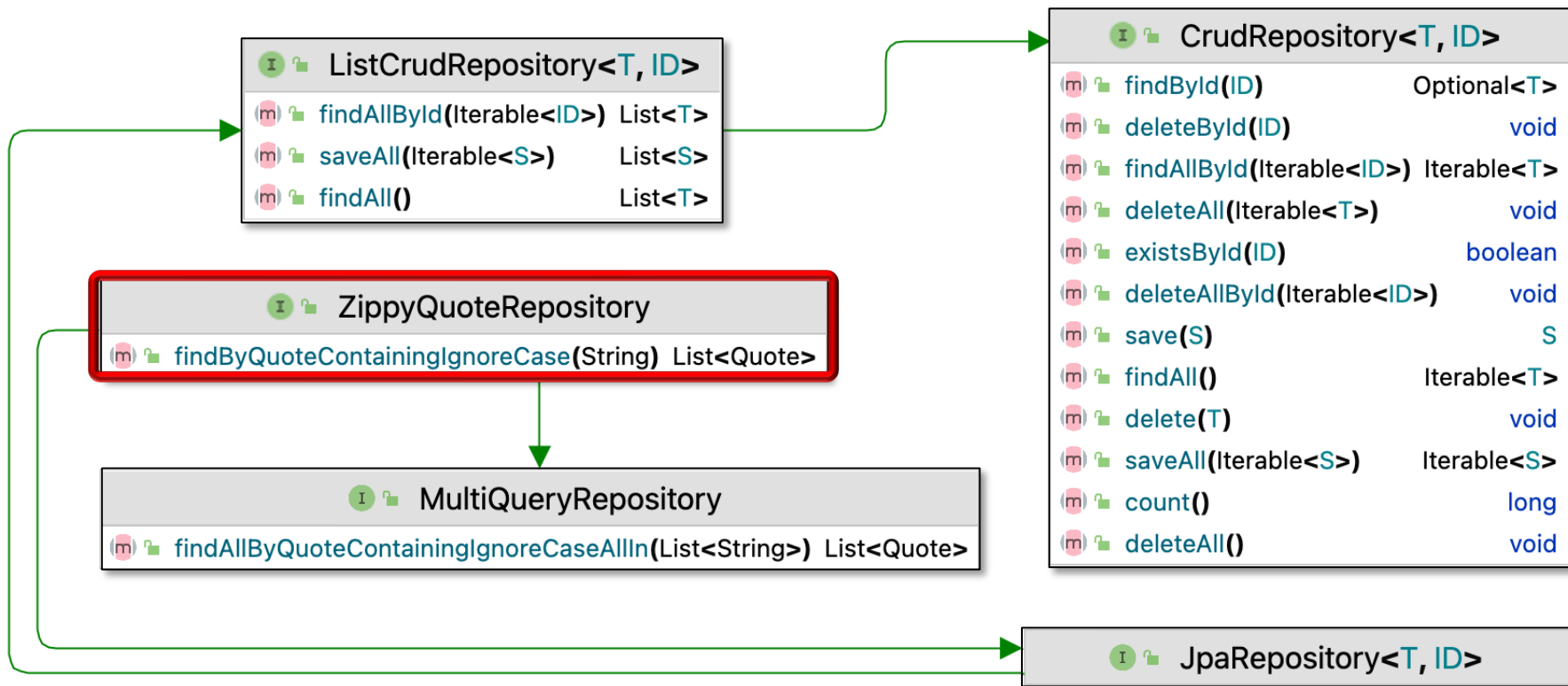
**Institute for Software
Integrated Systems**

**Vanderbilt University
Nashville, Tennessee, USA**



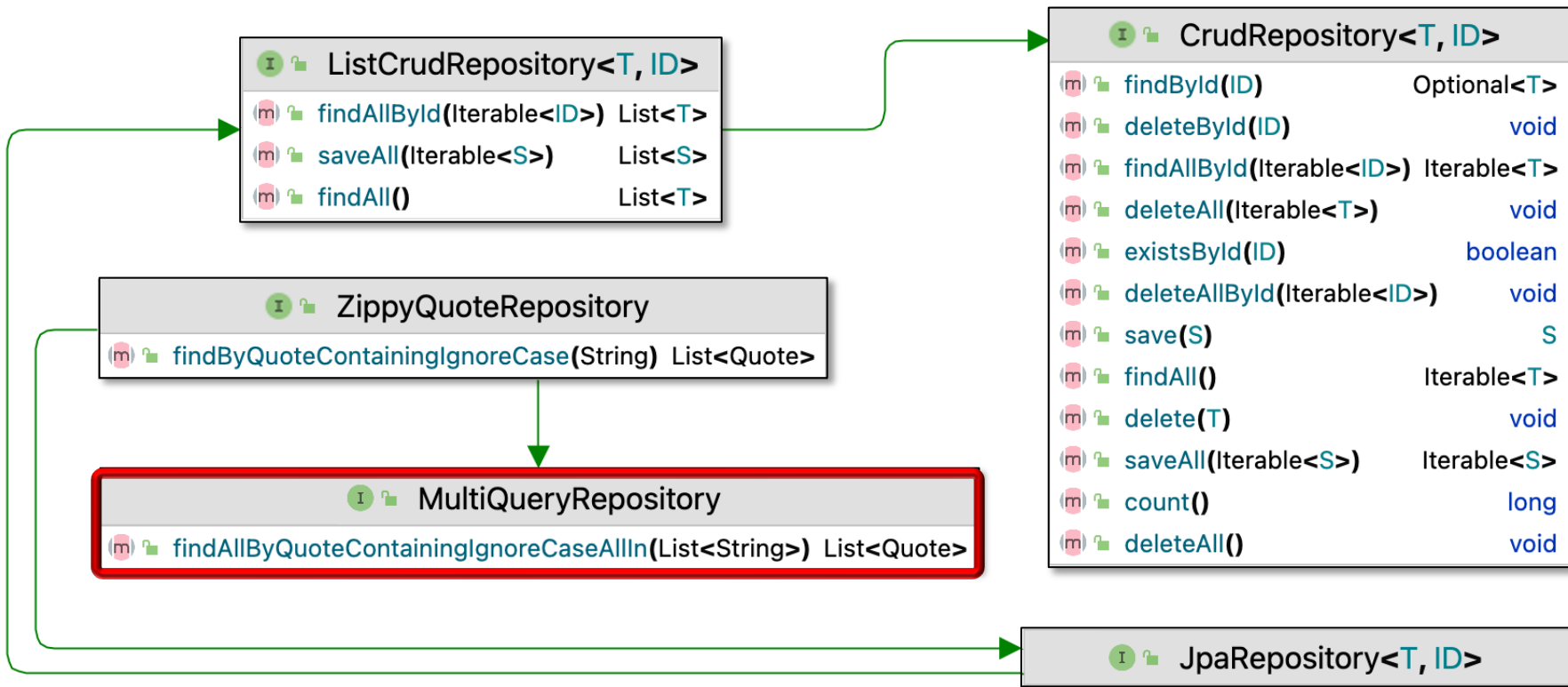
Learning Objectives in this Part of the Lesson

- Understand the implementation of the ZippyQuoteRepository



Learning Objectives in this Part of the Lesson

- Understand the implementation of the ZippyQuoteRepository



The main focus is on the custom SQL query in MultiQueryRepositoryImpl

Implementing the ZippyQuoteRepository

Implementing the ZippyQuoteRepository

```
17 public class MultiQueryRepositoryImpl
18     implements MultiQueryRepository {
19     /**
20      * This field represents a session with the database, providing
21      * the main API for performing CRUD (Create, Read, Update, Delete)
22      * operations and querying the database.
23      */
24     @PersistenceContext
25     private EntityManager entityManager;
26
27     /**
28      * Find a {@link List} of {@link Quote} objects in the database
29      * containing all of the {@code queries} (ignoring case).
30      *
31      * @param queries The {@link List} of queries
32      * @return A {@link List} of {@link Quote} objects in the database
33      *         containing at all of the {@code queries}
34      *         (ignoring case)
```

See github.com/douglasraigschmidt/LiveLessons/tree/master/WebMVC/ex4

End of the QuoteServices App Case Study: Implementing the Zippy Microservice (Part 1)