

# AirBnB (Clone) for Web Application Technology (UoA:DIT)

- **Functionality**

- This is the RESTful Web Service part of the project. It utilises the Jersey JAX-RS implementation.
- It uses a MySQL Database to store the data. More information can be found in the [Database](#) section of this file.

- **Implementation**

- Due to the size of the application, no implementation details are written here. These can be found in the form of JavaDoc in {ROOT}/jd/index.html

- **Database**

- The database is designed to be in 3NF, meaning that no redundant data is held in any of the tables and each reference to existing data is in the form of foreign keys.
- Accesses to the database are managed by Apache Commons' Connection pooling.

- **\_Tables**

Name	Usage	Key(s)	Foreign Keys	Additional Comments
bookings	Stores users' bookings	bookingID	userID(users), houseID(houses)	
comments	Stores each house's comments	commentID	userID(users), houseID(house)	This table is especially large. Thus, it's better to remove that section when exporting to XML
houses	Stores data about each registered house	houseID	ownerID(users)	
searches	Stores users' searches	searchID	userID(users), houseID(houses)	When we don't have enough information on a user's preferences, we keep track of house pages they've visited (later used in the sparse NNCF algorithm)
invalidTokens	Stores invalidated tokens	tokenID	None	When a user signs out, we add their token to the blacklist and consider it invalid
messages	Stores user to user messages	messageID	senderID(users), receiverID(users)	
photographs	Intermediate table that matches houses to photographs	photoID	houseID(houses)	
users	Stores user information	userID	None	Each user's password is hashed and stored in the database

- Third party libraries used:

Library	Version	Usage	Extra comments
<u><a href="http://jersey.github.io/">Jersey (http://jersey.github.io/)</a></u>	2.26-b09	The JAX-RS implementation of choice	
<u><a href="https://dev.mysql.com/downloads/connector/j/5.1.html">MySQLConnector (https://dev.mysql.com/downloads/connector/j/5.1.html)</a></u>	5.1.6	Connection with the MySQL Database	
<u><a href="https://commons.apache.org/proper/commons-pool/">Apache Commons Pool (https://commons.apache.org/proper/commons-pool/)</a></u>	1.4	Database Connection Pooling	Used instead of a JPA implementation
<u><a href="http://jersey.github.io">Jersey Media Json-Jackson (http://jersey.github.io)</a></u>	2.26-b07	Json Writer for jersey	
<u><a href="https://github.com/FasterXML/jackson">Jackson (https://github.com/FasterXML/jackson)</a></u>	2.9.0.pr4	Java JSON support	
<u><a href="https://jwt.io/">Json Web Token (JWT) (https://jwt.io/)</a></u>	0.7.0	Web Token Generator - Validator	
<u><a href="https://sites.google.com/site/gson/">Gson (https://sites.google.com/site/gson/)</a></u>	2.8.1	Google's JSON to POJO converter	
<u><a href="http://jersey.github.io">Jersey Media Multipart (jersey.github.io)</a></u>	2.17	Picture Uploads	
<u><a href="http://jamesmurty.com/2008/12/14/xmlbuilder-easily-build-xml-docs-in-java/">James Murty's XMLBuilder (http://jamesmurty.com/2008/12/14/xmlbuilder-easily-build-xml-docs-in-java/)</a></u>	1.1	Raw export of the database to XML	
<u><a href="https://projects.spring.io/spring-security/">Spring Framework Security (https://projects.spring.io/spring-security/)</a></u>	3.1.0.RELEASE	Password hashing	
<u><a href="http://jamesmurty.com/2008/12/14/xmlbuilder-easily-build-xml-docs-in-java/">James Murty's XMLBuilder (http://jamesmurty.com/2008/12/14/xmlbuilder-easily-build-xml-docs-in-java/)</a></u>	1.1	XMLBuilder for database to XML export	
<u><a href="https://commons.apache.org/proper/commons-csv/">Apache Commons CSVReader (https://commons.apache.org/proper/commons-csv/)</a></u>	1.5	Used to parse the provided CSVs and add the data to the database	
<u><a href="https://github.com/uniVocity/univocity-parsers">Univocity Parsers (https://github.com/uniVocity/univocity-parsers)</a></u>	1.0.0	Dependency of Apache Commons CSVReader	
<u><a href="https://github.com/tdebatty/java-LSH/tree/master/src/main/java/info/debatty/java/lsh">Debatty's LSH (https://github.com/tdebatty/java-LSH/tree/master/src/main/java/info/debatty/java/lsh)</a></u>	RELEASE	LSH Implementation	Used in the NNCF algorithm

