LCSS Report Team07

Lucas Dohmann, Jason Oltzen, Mert Torun, Luca Uckermann

Abstract—**This document describes the LCSS project of team 07. For our Lab project in the module Large and Cloud-based Software Systems we want to create a coupon application where a user can upload a coupon code and other users can up or down vote the coupon code, can comment and can search through hashtags. We primarily want to find out what impact in terms of performance different methods of getting data from a database and search through the results can have.**

Index Terms—**Cloud, Large-Scale system, Database models, Performance**

I. INTRODUCTION

TODAY, many companies are using coupons to attract customers and increase their sales. With the rapid growth of online shopping and e-commerce in recent years, coupon web applications are becoming increasingly more popular and important to reach a wider audience. Because of the large number of existing coupons, it is becoming more difficult for consumers to keep track. Therefore, it is important to provide an effective search method for users.

In our project, we want to investigate the effectiveness of different searching methods through the attributes of the coupons that are stored in a database. We want to compare the performance of relational and non-relational database management systems in terms of response times. Our research question is: Are there measurable and noticeable differences in different methods of searching through the coupons attributes inside the database, and what method performs the best? In terms of response times, is a relational or non-relational database the best option?

For our implementation of the coupon web application, we will be using the Python-based web framework Django and the Google Cloud Platform. Furthermore, we will collect data on the response times of different search methods and database management systems. Afterward, we will analyze the results to determine which method performs the best. With this project, we want to identify the most effective search methods, and how to optimize database performance in web applications.

In the following sections of this report, we will provide a review of the relevant literature, our prototype's system architecture, present our results, and discuss our measurements and findings.

II. RESEARCH QUESTION

A. Motivation

When you are working with databases, you will likely come in contact with the debate which is the most effective type of database management systems in terms of storing and managing the data. There are two main types: relational database management systems and non-relational database management systems. The choice of database management system can significantly impact the performance of a web application. Therefore, our project's focus is on investigating the effectiveness and performance of various database querying methods.

B. Formulation

Are there measurable and noticeable differences in different methods of searching through the coupons attributes inside the database, management system and what method performs the best? In terms of response times, is a relational or a non-relational database management system the best option?

C. Approach

To answer our research question, we will be using the Python-based web framework Django and the Google Cloud Platform. The application will allow users to upload a coupon, write comments under coupons, create hashtags, up and down vote a coupon and see the score of the related coupon (see figure 1). One main aspect of the application should be the search of a specific coupon based on for example a hashtag. We will implement both, a relational and non-relational database management system to store the coupon data and compare their performance in terms of response times for each search method.
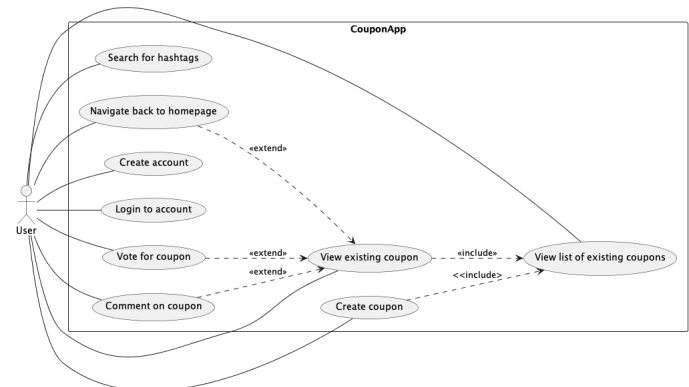

Fig. 1. UML Use Case diagram - Coupon App

To collect the data on the performance of the different search methods, we will use a performance testing tool like "Google Cloud Load Testing". With tools like this, we are able to simulate many users accessing the coupon web application and measure the response times for different search queries. After we have collected the data, we will compare the mean response times and standard deviations for each search method

and database management system and determine if there are significant differences in performance.

Our general approach is to develop a prototype coupon web application, test its performance using performance testing tools, and analyze the data collected from these tests to identify the most effective search methods and database management systems. By doing so, we aim to provide valuable insights for businesses looking to optimize their coupon web applications and improve the user experience for their customers.

III. Preliminary or Related Works

In this section, related work is presented. Our Project is inspired by the Mydealz web application [1] and we will implement a lightweight version of this application. Mydealz is an online platform where users can post and vote on deals, discounts and coupons from different branches.

In the article "Enhanced query processing over semantic cache for cloud based relational databases" [4] published in April 2020, the authors proposed the approach for improving the hit rate of the cache system by using an implicit semantic matching algorithm that involves four different scenarios, where each is designed to optimize the cache hit rate based on the query predicate and attribute matching. Furthermore they evaluated different experiments, which showed that it outperformed existing techniques in terms of time complexity and hit rate. The experiments were conducted on a prototype system that included a database management system and were cache maintained on the same device.

The paper "Integrated Environment Based on Anytime Solution Search Algorithms and A Non-Relational Database for Real-Time Intelligent Systems" [3] describes an integrated environment for real-time intelligent systems that uses anytime solution search algorithms and a non-relational database. The use of anytime algorithms and NoSQL databases improves efficiency when working with graph-based and speeds up the process of obtaining a solution.

The article "BANKS: Browsing and Keyword Searching in Relational Databases" [2], discusses Browsing And Keyword Searching (BANKS), which allows for easy publishing of relational and XML data on the web. Relational database management systems require knowledge of schema and query language for a query that is used to get specific information from a database when the need arises, which makes it difficult for casual users to make search requests. An alternative would be an HTML form, where the users can use predefined queries, but creating forms for each task is laborious and confusing. The best solution would be to create Search engines on the web, that offers a simpler alternative through unstructured querying and browsing using keywords and hyperlinks. Keyword search can provide a user-friendly mechanism for casual users to access database information without requiring schema knowledge.

References

[1] Mydealz. hhttps://www.mydealz.de/. Accessed: 2023-04-26.
[2] Soumen Chakrabarti Arvind Hulgeri Charuta Nakhe Parag S. Sudarshanxe B. Aditya, Gaurav Bhalotia. Banks: Browsing and keyword searching in relational databases. https://www.sciencedirect.com/science/article/pii/B9781558608696501141. Accessed: 2023-04-26.
[3] Paniavina Eremeeva, Poliushkina. Integrated environment based on anytime solution search algorithms and a non-relational database for real-time intelligent systems. https://ceur-ws.org/Vol-2648/paper26.pdf. Accessed: 2023-04-26.
[4] Atta Rahman Rachid Zagrouba Fahd Alhaidari Tariq Ali Farzana Zahid Munir Ahmad, Muhammad Abdul Qadir. Enhanced query processing over semantic cache for cloud based relational databases. https://link.springer.com/article/10.1007/s12652-020-01943-x. Accessed: 2023-04-26.