

ABSTRAK

Pituin *coffee* masih menggunakan inventori konvensional hal ini akan menyebabkan ketidakefisienan dalam pendataan dan juga inventori konvensional sudah ketinggalan zaman dan akan menghambat proses untuk kedepannya. Pengelolaan inventori secara konvensional juga menjadi masalah yang mengakibatkan pemborosan biaya pada kertas yang akan dipakai.

Tujuan dari penelitian ini adalah untuk memahami rancangan Database Life Cycle yang dapat memenuhi kebutuhan sistem manajemen inventori, memungkinkan data disimpan, diakses, dan dikelola dengan efisien. Selain itu, penelitian ini bertujuan untuk merancang model database relasional yang sesuai dan efektif guna mendukung manajemen inventori serta mengatasi perubahan yang terjadi pada inventori.

Database Life Cycle adalah pendekatan yang menjelaskan tentang siklus kehidupan dari pangkalan data. Siklus dalam DBLC berulang secara berkesinambungan karena pangkalan data mengalami perubahan sesuai dengan perkembangan kebutuhan yang ada. Proses inti dalam merancang sebuah pangkalan data dapat dibagi menjadi tiga tahap, yakni perancangan skema konseptual, perancangan logika, dan perancangan fisik.

Hasil pengujian basis data menunjukkan potensi dalam menghasilkan data akurat secara individual saat melakukan pengambilan data, yang secara khusus memenuhi kebutuhan sistem manajemen inventori. Proses perancangan database menggunakan metode Database Life Cycle telah menghasilkan bentuk database relasional dengan langkah-langkah berikut: Pertama, pada tahap Conceptual Database Design, diperlukan pembentukan dua belas entitas dengan penentuan domain atribut untuk setiap entitas, dan ini menghasilkan diagram hubungan entitas. Selanjutnya, pada tahap Logical Database Design, relasi dibangun untuk merepresentasikan model data logis yang mencakup entitas, hubungan, dan atribut yang telah diidentifikasi sebelumnya. Terakhir, pada tahap Physical Database Design, desain pangkalan data diimplementasikan menggunakan MySQL sebagai sistem manajemen pangkalan data, dengan DDL digunakan sebagai bahasa untuk mendefinisikan struktur data.

Kata kunci: Pituin *Coffee*, Manajemen Inventori, DBLC

ABSTRACT

Pituin Coffee still relies on konvensional inventory management, which leads to inefficiencies in record-keeping and is outdated, potentially hindering future processes. Konvensional inventory management is also problematic, resulting in cost wastage on paper usage.

The aim of this study is to comprehend the Database Life Cycle design that can fulfill the requirements of an inventory management system, allowing data to be stored, accessed, and efficiently managed. Furthermore, the research aims to design a suitable and effective relational database model to support inventory management and address changes occurring within the inventory.

The Database Life Cycle is an approach that outlines the lifecycle of a database. The DBLC cycle is iterative due to the continuous changes that databases undergo based on evolving needs. The core process of designing a database can be divided into three stages: conceptual schema design, logical design, and physical design.

The results of the database testing exhibit the potential to generate precise individual data during data retrieval, specifically meeting the demands of the inventory management system. The database design process, using the Database Life Cycle methodology, has yielded a relational database with the following steps: Firstly, in the Conceptual Database Design stage, the creation of twelve entities with the specification of attribute domains for each entity is essential, resulting in an entity relationship diagram. Subsequently, during the Logical Database Design phase, relationships are established to represent the logical data model encompassing entities, relationships, and previously identified attributes. Lastly, in the Physical Database Design phase, the database design is implemented using MySQL as the database management system, with Data Definition Language employed to define the data structure.

Keywords: Pituin Coffee, Inventory Management, DBLC