



ARCHITECTURE DOCUMENT

With C4 Architecture



APRIL 20, 2021

FONTYS - UNIVERSITY OF APPLIED SCIENCES - ICT
Ebowusim Michael - 4050630

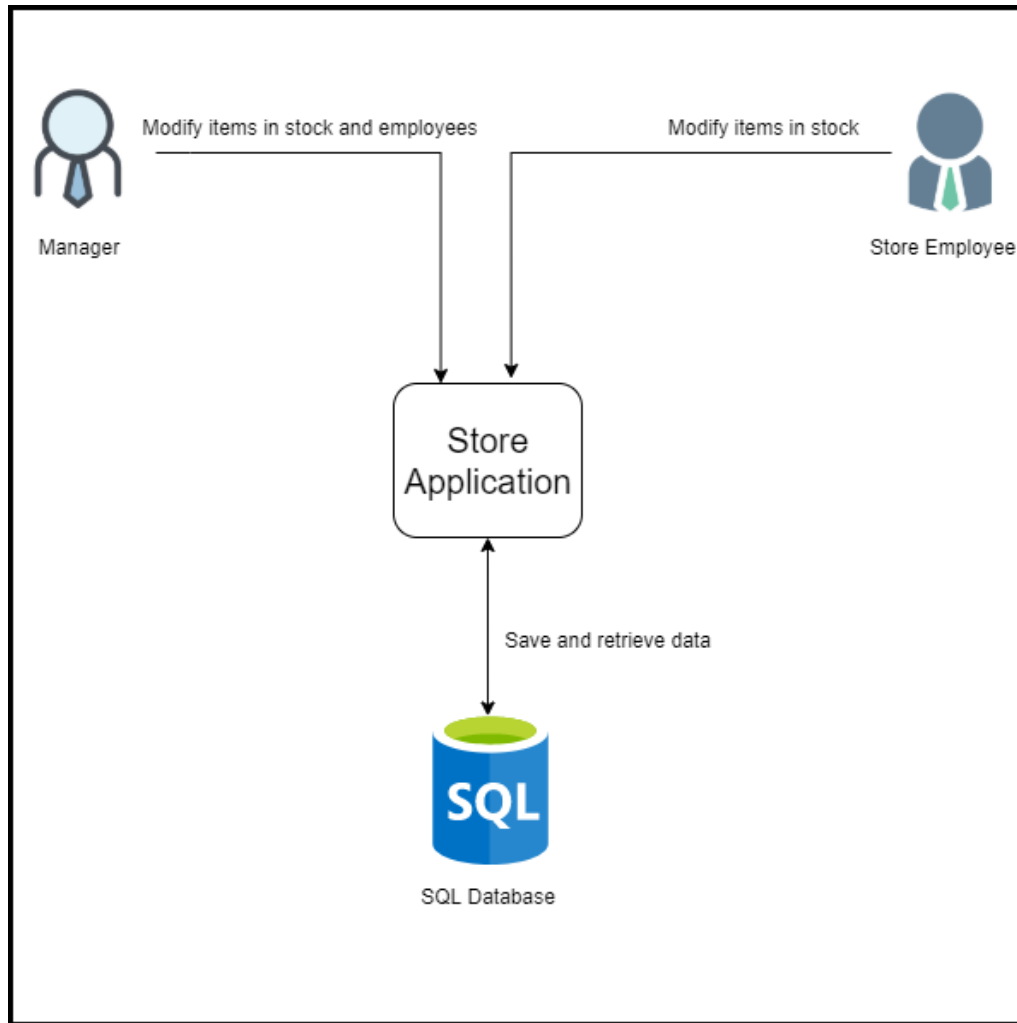
Version Control

Version	Changes
V0.1	Added C4 diagram and explanation.
V0.2	<ul style="list-style-type: none">Improved the explanation and diagrams of the other models.Added version control.

Table of Contents

1	System Context (C1 model)	1
2	Containers (C2 model)	2
3	Components (C3 model)	Error! Bookmark not defined.
4	Class diagram(C4 model)	Error! Bookmark not defined.

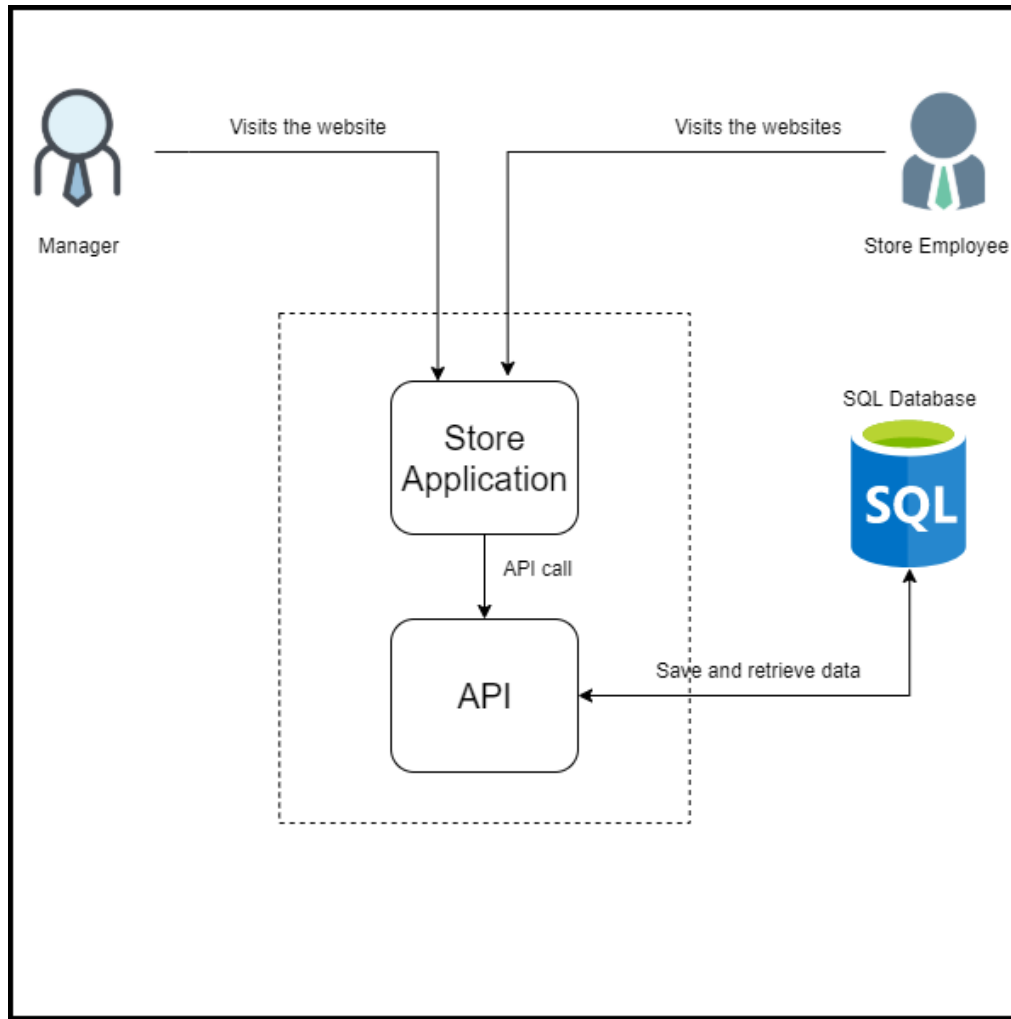
1 Context



In this diagram, you see two users. The Manager who will use the application by inputting/changing data e.g adding items data to the stock , as well as adding/removing employees under him. The Employee who is on a lower level than the manager and can only add/modify/remove data in the application.

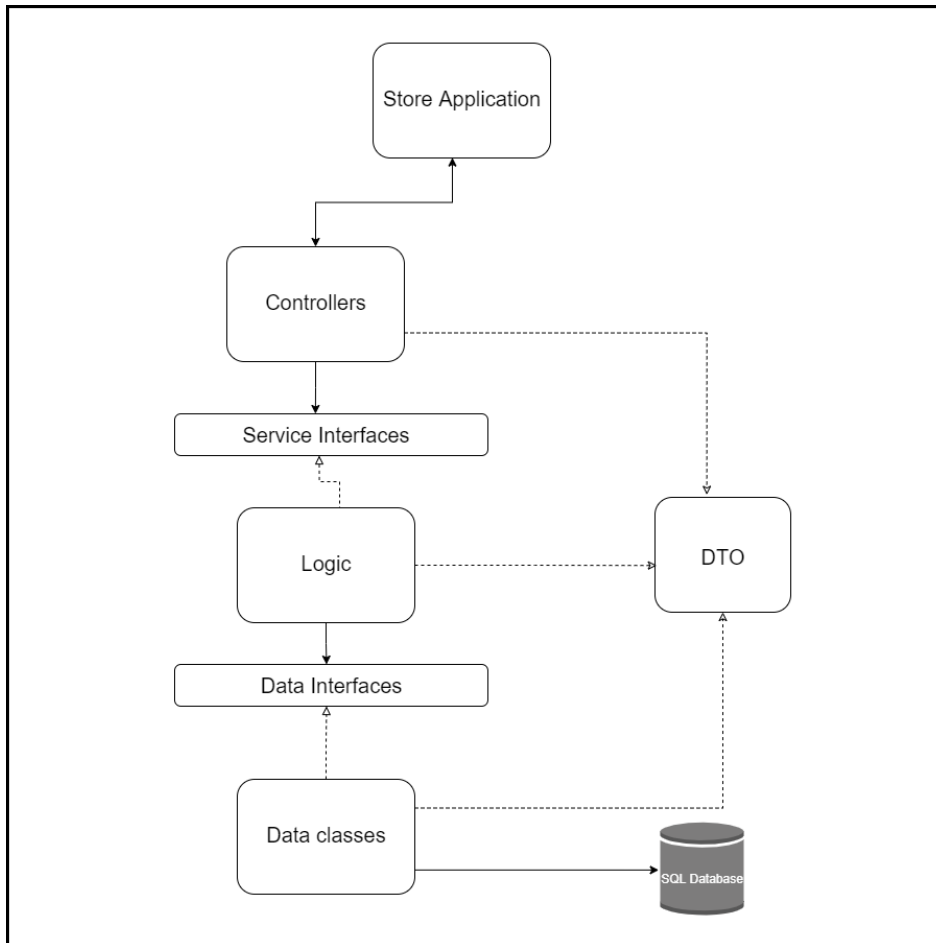
There is a database using the Hera server which is seen as an external system that is used to store and retrieve data.

2 Containers



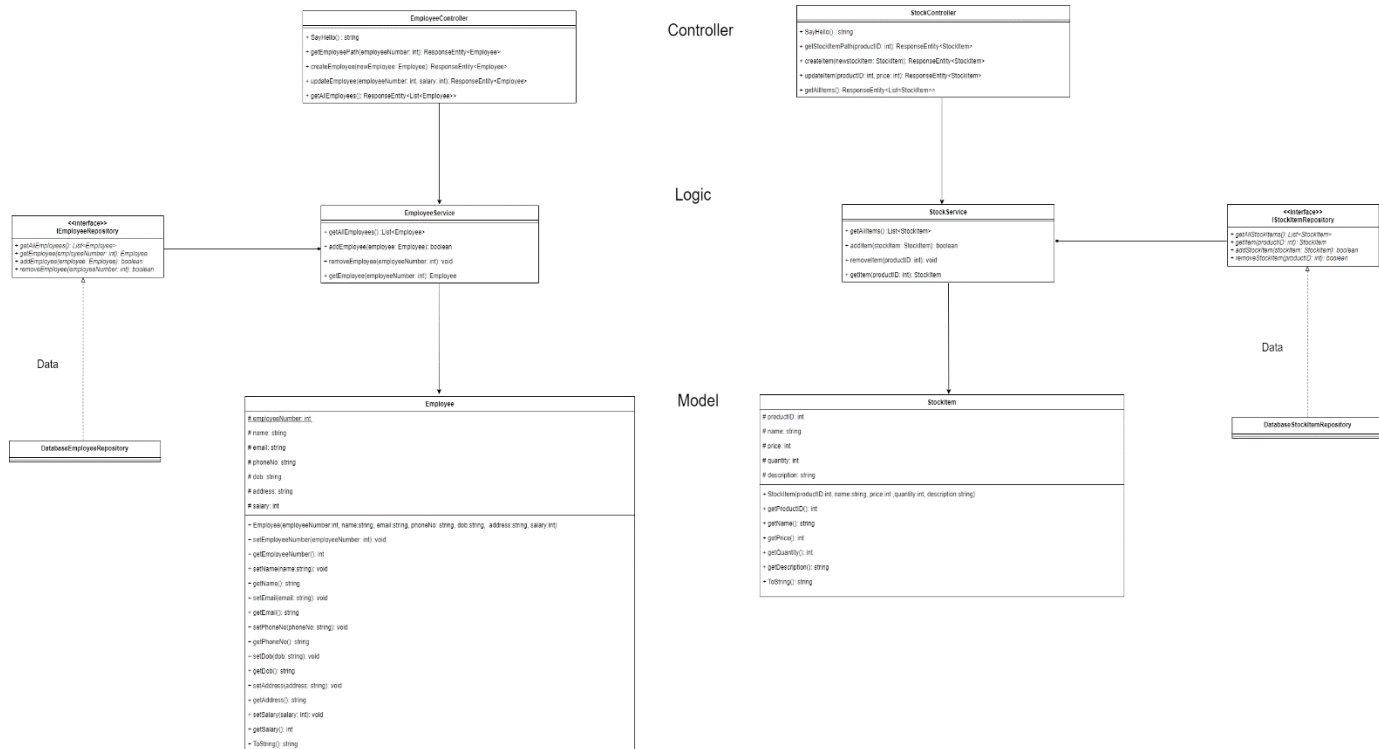
The diagram above is a more focused version of the context(C1 model). This application consists of 2 containers. The website made using React framework and Javascript and the API which was made with Springboot and Java. The Manager visits the website where he inputs/modifies data. The web application makes a call to the API which would store/ retrieve the requested data via the database. This is the same process for the Employee when using the web application.

3 Components



In this diagram above, the classes are divided into certain layers for a better improved and enhanced application that goes with the **SOLID** principles. The **Controllers** will receive and return the request call from the front-end. The **Logic** classes will do the logic and the algorithm functions. **Data** classes will store and retrieve data to/from the database. They are the only layer that has access to the database. **DTOs** (Data transferable objects) are used to send data to each layer. The Data classes and Logic classes each inherit from their interface layers. This is all to follow the SOLID principles and make the application a more flexible one.

4 Class Diagram



In the Employee model class, the date variable is declared as a String instead of Datetime. This is to avoid errors in the front-end .

The Controller classes(`EmployeeController`, `StockController`) use Create,Get and Update functions which will help to add/modify/delete data in the application.

The Logic classes(`EmployeeService`, `StockService`) use Add,Get and Remove functions which will also help to add/delete data in the application. There is no separate Update function because the `save()` from the `JpaRepository` does both the Create/Add and Update functions already.