# Architecture Design Document for BooXChange

Software Engineering Course, Sem 6
Prepared on 22nd April 2020

PREPARED FOR
Prof. Khushru Doctor

Version 1.0

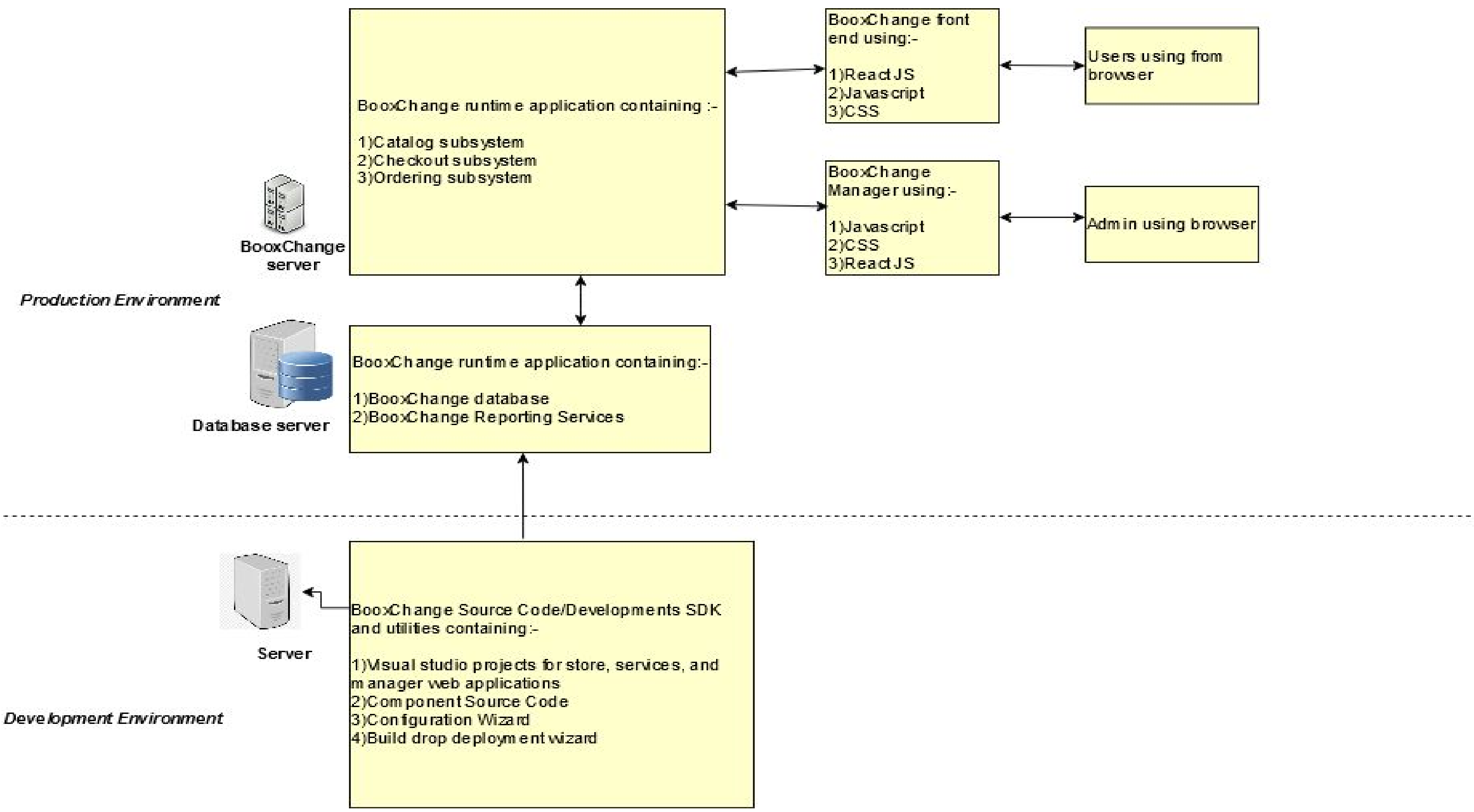
Prepared By :

| | |
|---|---|
| Jainam Chhatbar - 1741002 | Aman Dave - 1741003 |
| Harshiv Patel - 1741005 | Aditya Shah - 1741007 |
| Dhairya Dudhatra - 1741058 | Meet Modi - 1741071 |
| Shantanu Sheth - 1741088 | |

# Table Of Contents

# 1. Architecture

## 1.1 Technical Architecture

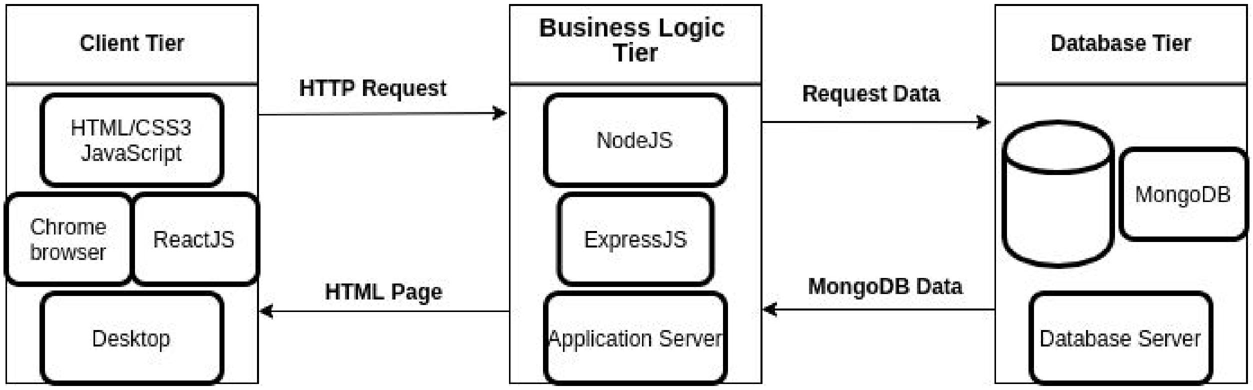

BooxChange runtime application containing :-
1)Catalog subsystem
2)Checkout subsystem
3)Ordering subsystem
BooxChange front end using:-
1)ReactJS
2)Javascript
3)CSS
Users using from browser
BooxChange Manager using:-
1)Javascript
2)CSS
3)ReactJS
Admin using browser
BooxChange server
Production Environment
Database server
BooxChange runtime application containing:-
1)BooxChange database
2)BooxChange Reporting Services
Server
Development Environment
BooxChange Source Code/Developments SDK and utilities containing:-
1)Visual studio projects for store, services, and manager web applications
2)Component Source Code
3)Configuration Wizard
4)Build drop deployment wizard

**BookXchange Technical Architecture**

The above figure depicts the whole software development cycle for product BookXchange. Here there are two environments considered in this design :

- **Development Environment** includes the whole tech-stack on which the code has been developed using developer tools like IDE and Databases and all the configuration of various parameters have been done. This environment contains many version-controlled codes which are buggy and in not working condition. This environment includes only development server and is only meant for development use only.
- **Production Environment** includes the final release or version controlled release of the software which is meant for use of general public and the organization. This Environment incorporates business application needs and has a database server that will be used by businesses and the organization, also there will be a web server which will host the whole Application on the web.

## 1.2 System Architecture

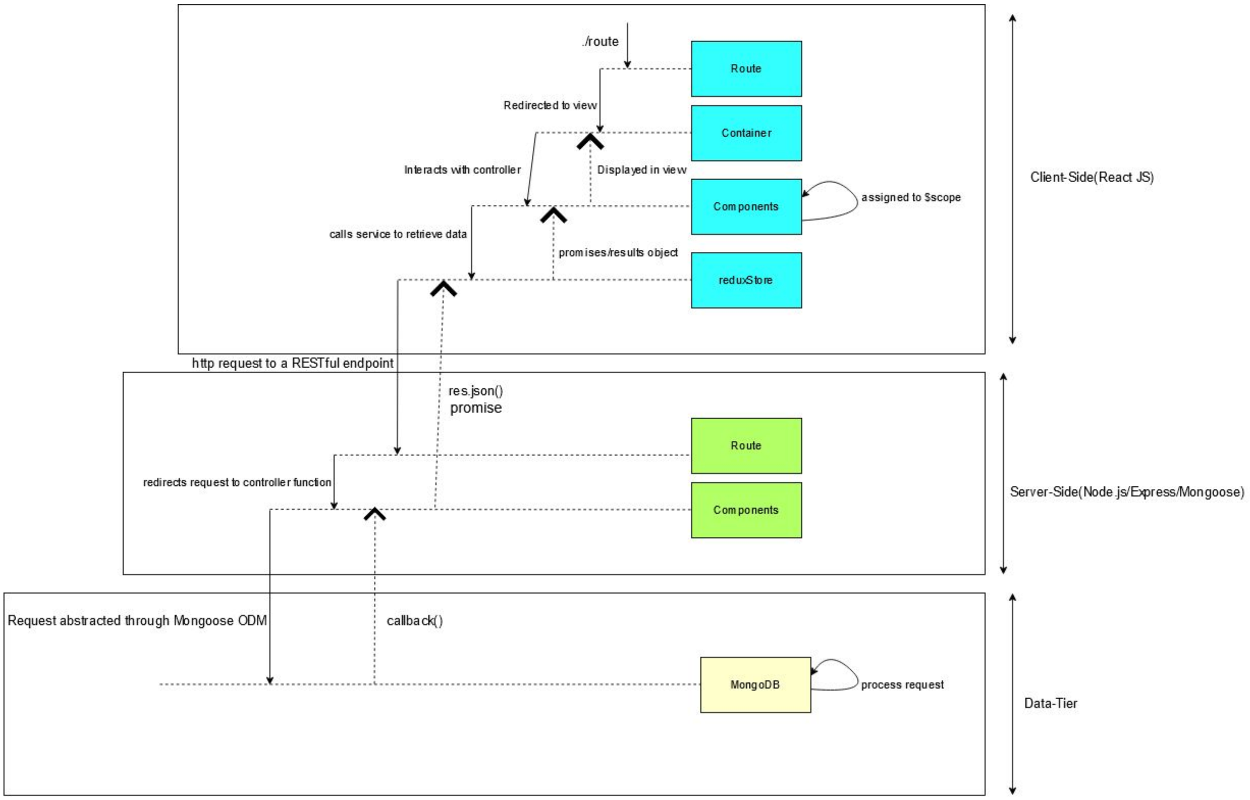


Here, the design of BookXChange Software is done in consideration of NFR's stated in the above chapter 2.We have proposed using NoSQL architecture, for faster development of the product.BookXchange product will be using the MERN (MongoDB,Express.js, React, Node) for development.

- **MongoDB:** MongoDB is a document database, which means it stores data in JSON-like documents.This has a great advantage for handling data in JSON objects and these data objects can be transferred throughout to the front-end without any interference or conversion through different technologies.
- **Express.js:** It is a modular web framework for Node.js. It makes it easier to write secure, modular and fast applications
- **React.js:** It enables developers to create web applications in a scalable, simple and fast manner
- **Node:** Easier way to write back-end APIs and handle the server.

The team was familiar with Javascript, and so we all decided to stick with MERN stack.

## 1.3 Architecture Process


./route
Route
Redirected to view
Container
Interacts with controller
Displayed in view
Components
assigned to $scope
calls service to retrieve data
promises/results object
reduxStore
Client-Side(React JS)
http request to a RESTful endpoint
res.json()
promise
Route
redirects request to controller function
Components
Server-Side(Node.js/Express/Mongoose)
Request abstracted through Mongoose ODM
callback()
MongoDB
process request
Data-Tier