LoanWell

In-house lending platform management system

Batch 2 - Group 3

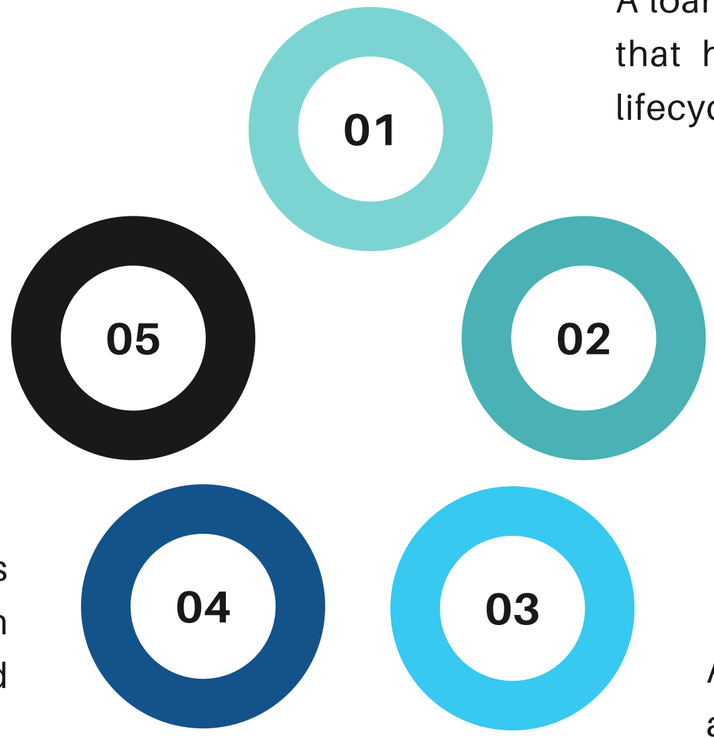
Chinmay Joshi, Parva Patel, Hariom Vyas
Samyak Jain, Shaurya Shekhar

INTRODUCTION

An overview of loan management systems

A loan management system is a digital platform that helps automate every stage of the loan lifecycle, from application to closing.

Lending applications that have a wide range of use cases, will surely attract more users than apps targeting only one specific loan type.



The traditional loan management process is meticulous and time-consuming.

Every stage of the lending process involves working with customer data. The best loan servicing software stores this data in centralized storage accessible during every processing stage.

A loan servicing software not only automates these procedures but also provides useful analytics and insights for lenders and borrowers.

OBJECTIVES

What are our key takeaways & learnings through this LMS

Flexibility

Employees can choose items and categories they want a loan for

Layered Approach

Must follow principles of a layered model, view and controller (MVC) architecture

Customer Management

Customers and their data being stored and utilized for writing functions

SDLC with CI-CD

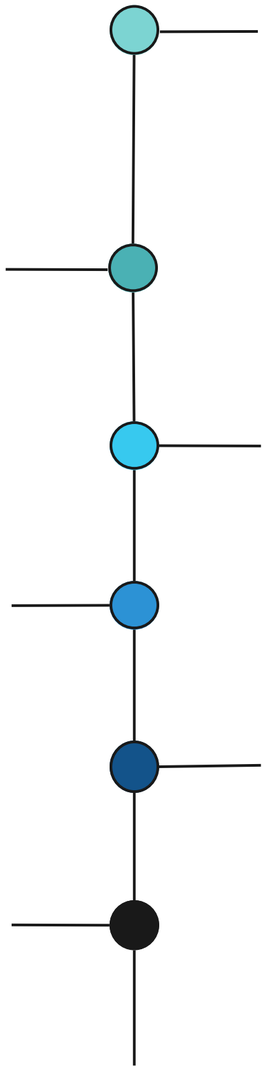
To get an understanding of the software development life cycle using continuous integration and continuous deployment

Agile Processes

Developing and ideating the project based on circumstances and needs

User Friendly Approach

Making simple and understandable user interfaces using React.js front end



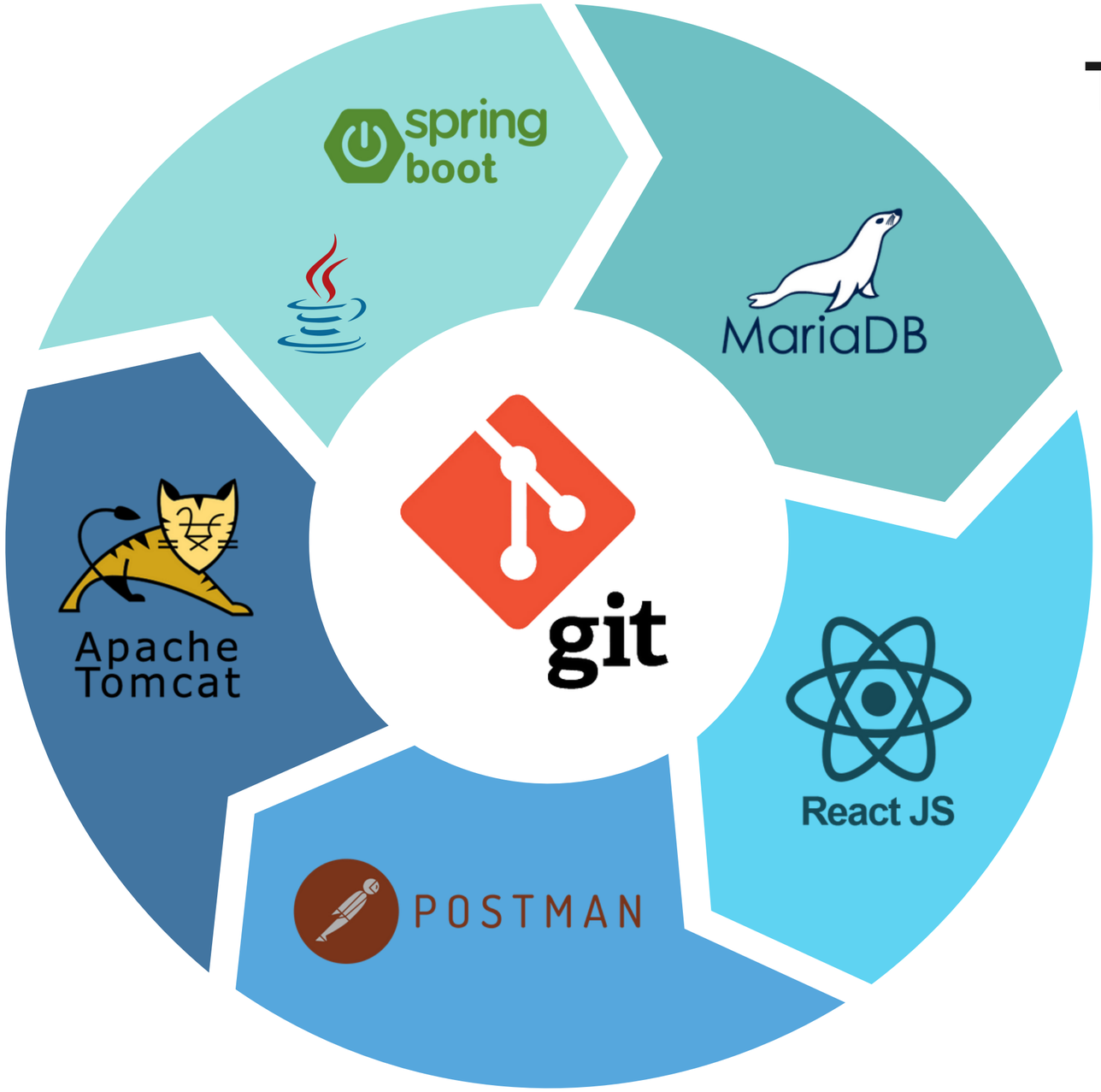
FUNCTIONALITIES OF LOANWELL PROJECT

What tasks can our web-application do?

- Maintains Login credentials providing security and differentiated page view for different users.
- Applying and tracking paying back of loans, ensuring that the employees can chart out a manageable way to pay back the loans in time.
- Notifying the employees whether the loan is overdue or not. This generates alacrity among the employees to pay back the loans on time.
- Displaying loan details and monitor/update the same. Hence empowering the admin and the employees to claim non-repudiation in fiscal matters.
- Recording the details of the Item being loaned out to establish a clear line of credit for the lender and for the employee.
- A responsive and interactive UI to accompany the backend functionalities to deliver a promising and user friendly experience.

USER AND ADMIN CHARACTERISTICS

WHAT CAN THE USER DO?	WHAT CAN THE ADMIN DO?
View his Purchased Items	**CRUD for Employee**
View His Loans Applied	CRUD for Item, Loan
Apply For A Loan	Reject a Loan
Login to the Portal	Sanction a Loan
Reapply For same item when he was earlier rejected	Declare if employee has paid back the loan



TECHNOGOLOGY STACK

Tools & technologies we used in our project

Java & Spring Boot

Formed the basis of Backend work, allowed implementation of CRUD and other functions

MariaDB

Provided a visual representation of the database

React.js

Undergirded the Frontend work, enabled the use of SPA (Single-Page Application) for project.

Git / GitHub

Version Control System

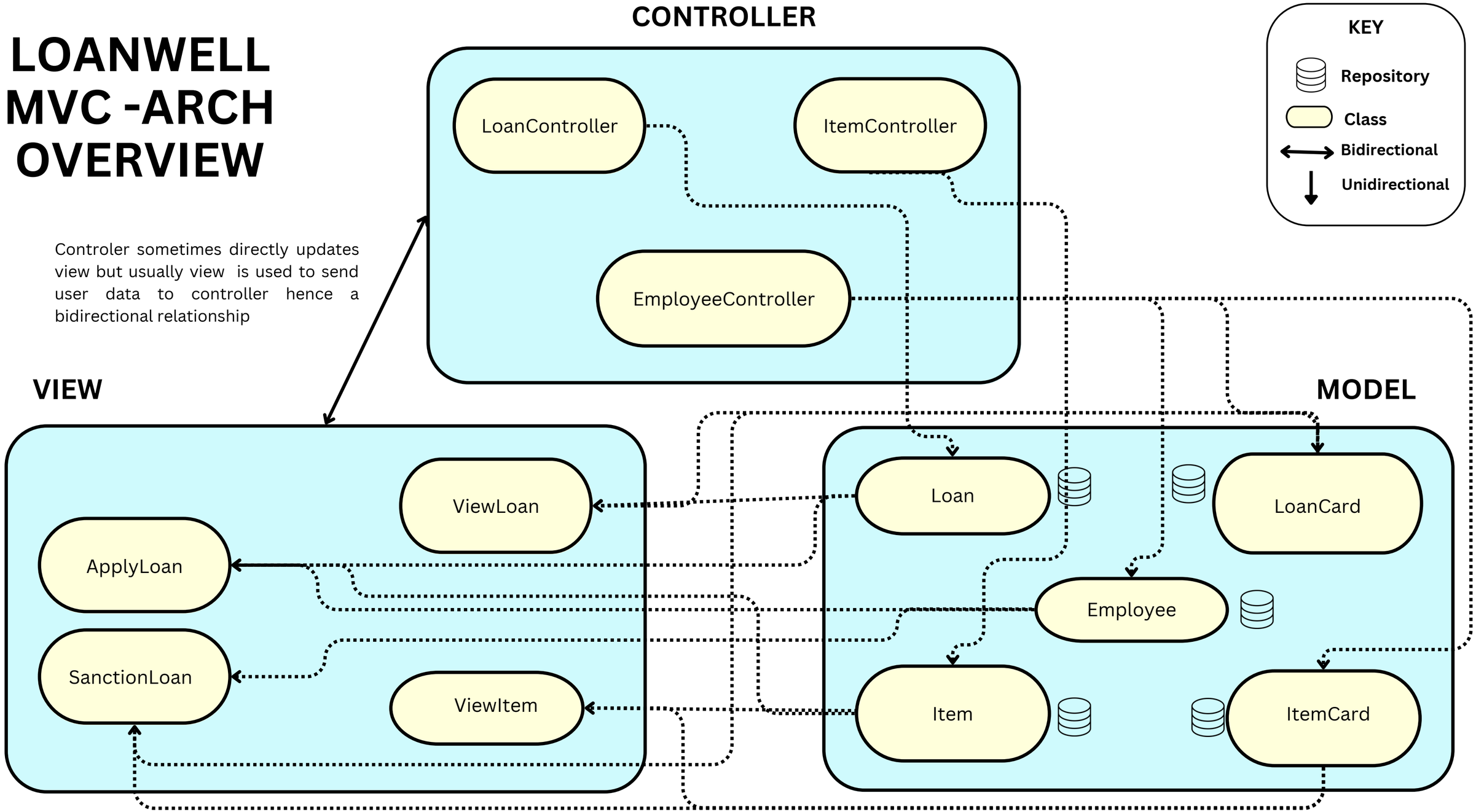
Postman

Enabled the constant REST API testing during the creation of the project.

Apache Tomcat

Provided a web server environment

LOANWELL MVC -ARCH OVERVIEW



TESTING AND EXCEPTION HANDLING

Testing and exception handling forms a quintessential part of any software life cycle. To enable a smoother and more robust development cycle, we have ensured commensurate testing measures, ranging from unit testing all the way to custom exceptions in the backend and also integration tests to ensure relative frontend and backend interactions.

We implemented Junit tests to test the functionalities of the Controller classes. Then we also went ahead and enabled the use of inbuilt and custom exceptions to showcase the errors of the applications.

There were 3 controllers - Employee, Loan and Item
All CRUD APIs were tested for all controllers. For employee, login and register functionality were also tested.

File | C:/Users/Administrator/Desktop/loan-well-final/index.html

java (29-Aug-2023 11:01:04 pm) > loan-well-final

Sessions

loan-well-final

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
src/main/java		39%		41%	154	232	250	456	142	220	5	21
src/test/java		100%		n/a	0	34	0	233	0	34	0	4
src/main/resources		n/a		n/a	0	0	0	0	0	0	0	0
target/generated-sources/annotations		n/a		n/a	0	0	0	0	0	0	0	0
target/generated-test-sources/test-annotations		n/a		n/a	0	0	0	0	0	0	0	0
Total	1,043 of 3,165	67%	14 of 24	41%	154	266	250	689	142	254	5	25

TEAM - LOANWELL

Batch 2 - Group 3

CHINMAY JOSHI	PARVA PATEL	HARIOM VYAS	SAMYAK JAIN	SHAURYA SHEKHAR
Creating clean back end code structure on Spring boot	Designing UI flow on React and continuous updates on it	Desiging UI flow on React and continuous updates on it	Writing certain service layer and controller layer functions	Native SQL queries on the backend
API Testing using Postman after writing code	Collaboration with backend to integrate, make fields w.r.t. to that	Making the UI responsive and providing an easy to understand and user-friendly experience	Preparing Models and writing Queries for fetching data from DB	Designing the Views as per the Wireframe requirements
Handling necessary dependencies and dealing with deprecations for diff tasks	Components update w.r.t. values fetched from the user/admin	Continuous-integration, continuous updating	Testing and debugging the code to achieve a 80 % coverage	Exception handling and some custom exceptions
Presentation Work	Presentation Work	Presentation Work	Presentation Work	Presentation Work

FUTURE SCOPE

Which extra features can we add in our project

Increasing computing

Get credit scores based on past history

Incorporating vernacular UI

Making the UI more approachable

Incorporating collaterals

To increase the system’s recovery rate

Adding payments module

Allow user to pay using this system itself

Cloud access

Deploying the system on the cloud