

# **TREASURY Bank ATM system**



# Contents

<b>Introduction</b>	<b>4</b>
<b>Functional Understanding and Description</b>	<b>4</b>
<b>Tech Stack</b>	<b>5</b>
Client-Server	5
Web server	5
Data Server	5
<b>Application Architecture</b>	<b>6</b>
<b>Database Model</b>	<b>7</b>
<b>User Management</b>	<b>8</b>
<b>Basic UI structure</b>	<b>8</b>

## PROBLEM STATEMENT

“In this covid stricken society,customers of TREASURY Bank need to withdraw money and see the savings balance without contact with the ATM machine”

# Introduction

Treasury is a website for an ATM system which aims to help Treasury bank customers to withdraw money from their bank deposit using their full name and sufficient withdrawal amount. After the withdrawal, the application will print the savings balance on the screen. If the entered credentials are wrong, then the application will also generate alert messages regarding the same.

## Functional Understanding & Description

The Treasury website contains a responsive User Interface(UI).

- User Interface: Contains 2 input fields for username and amount to be withdrawn. It also has a 'Withdraw' button to withdraw the cash.
- The credentials and the amount is checked against the database in the server.
- Withdrawal request: If the credentials are correct and the withdrawal amount is not higher than the savings balance, the savings balance is printed on the screen along with a 'Withdrawal Successful' message, on clicking the 'withdraw' button.

If the credentials are incorrect or the withdrawal amount is higher than the savings balance, the UI will generate an alert message box on clicking the 'withdraw' button.

# Tech Stack

- FRONT END - HTML5,CSS,JavaScript,Bootstrap
- BACK END -MySQL,PHP,Node js

## Client-Server

- The client-side of the webpage is done using HTML5, CSS, JavaScript .

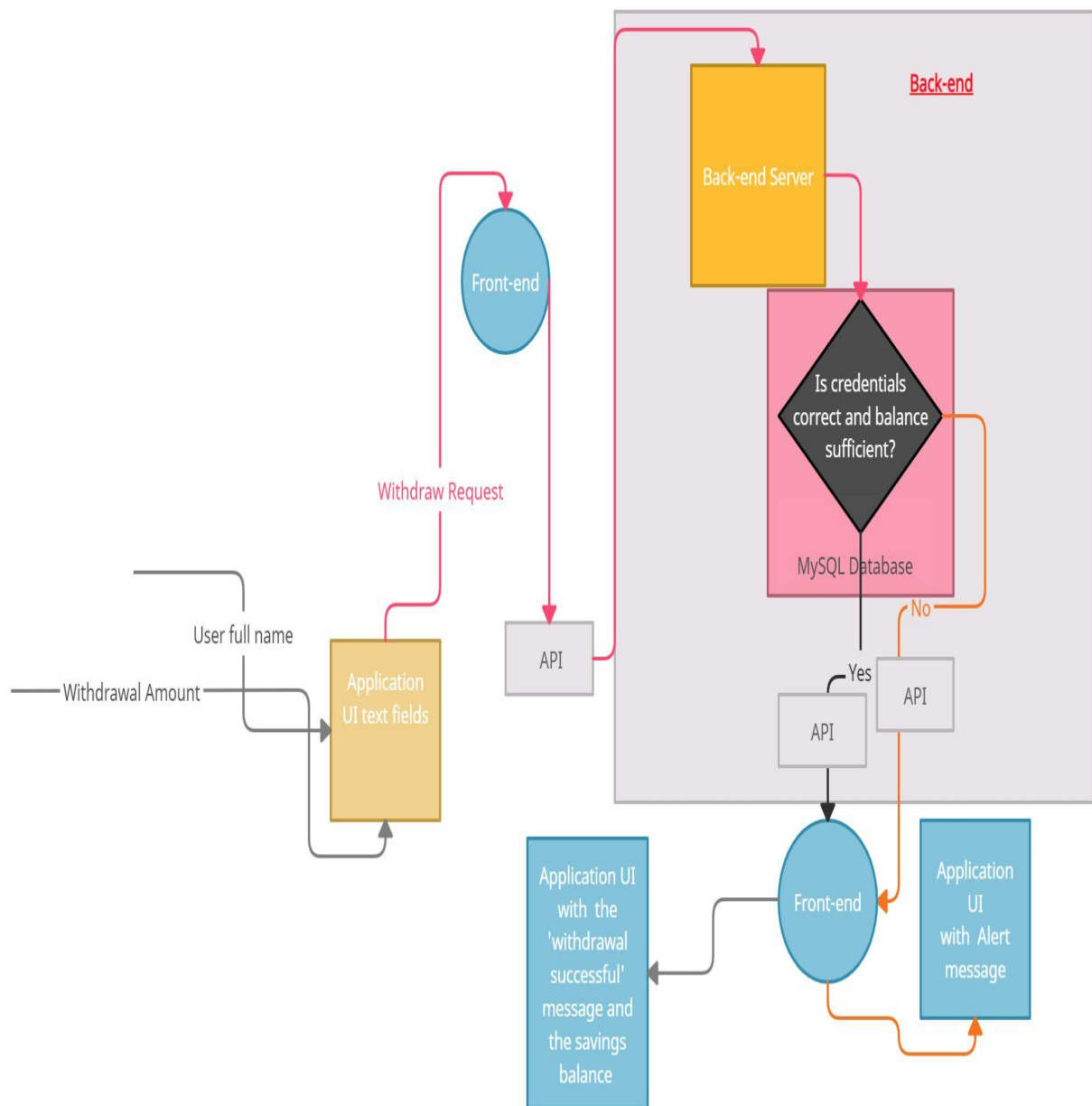
## Web server

- Back end Server setup by node js and express framework

## Data Server

- MySQL was used for managing the database of the application.
- MySQL is used for its RDBMS structure and ease of data fetching from different tables.

# Application Architecture



# DATABASE MODEL

```
mysql> describe users;
```

Field	Type	Null	Key	Default	Extra
username	varchar(30)	YES		NULL	
balance	float	YES		NULL	

```
2 rows in set (0.31 sec)
```

```
mysql> select * from users;
```

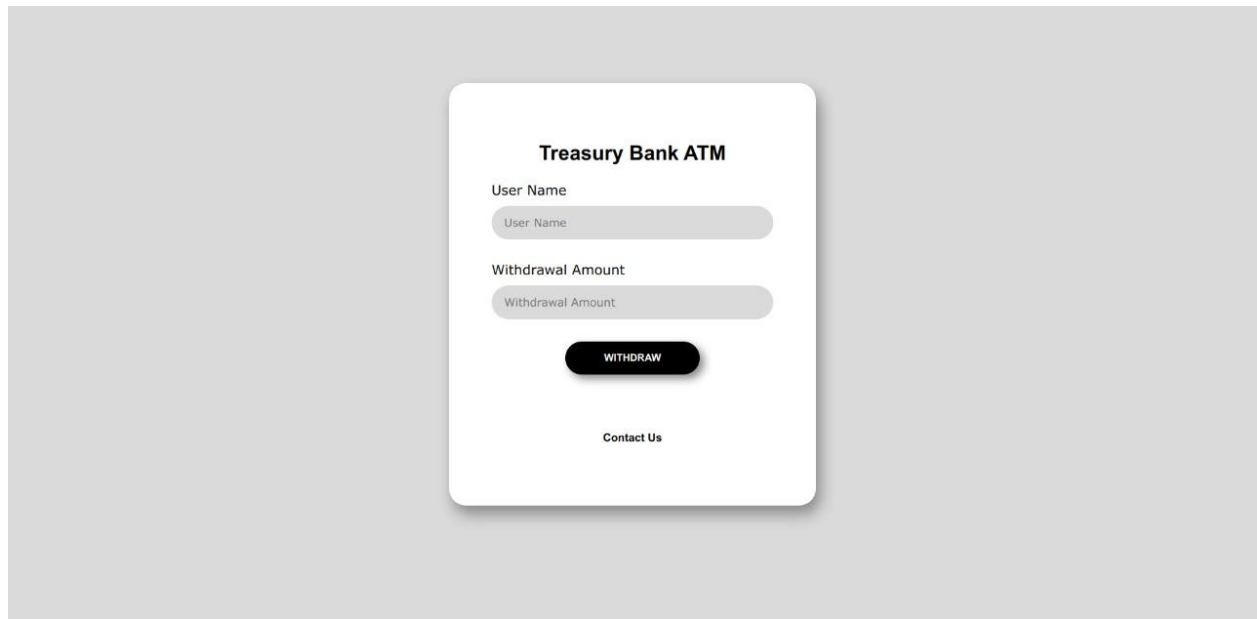
username	balance
Devi	10000
Dayana	20000
Rashad	30000
Imthiyas	40000
Jeevan	50000

```
5 rows in set (0.00 sec)
```

## User Management

Users will get their access to the application after they enter correct credentials and withdrawal amount must be less than savings balance. Thus they can withdraw money. If the credentials are incorrect or withdrawal amount is greater than savings balance, they will get an alert message.

# Basic UI Structure



Treasury Bank ATM

User Name

User Name

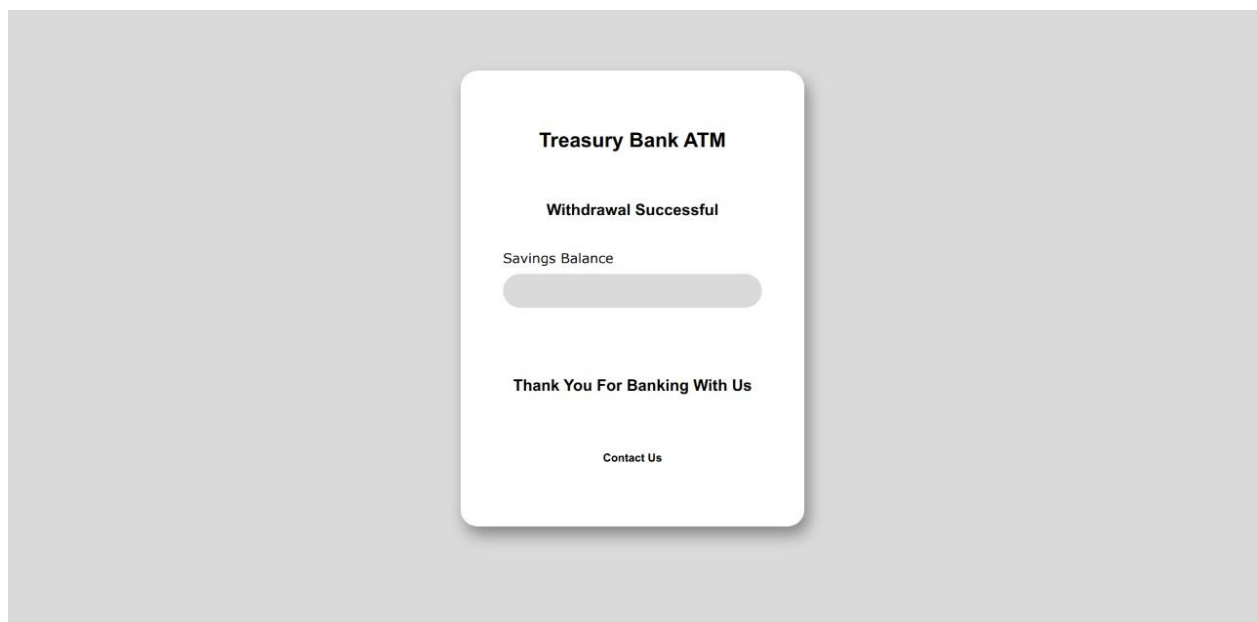
Withdrawal Amount

Withdrawal Amount

WITHDRAW

Contact Us

This is a wireframe of a Treasury Bank ATM withdrawal screen. It features a white rounded rectangle centered on a light gray background. The screen displays the title 'Treasury Bank ATM', followed by two input fields for 'User Name' and 'Withdrawal Amount'. Below these is a black 'WITHDRAW' button and a 'Contact Us' link at the bottom.



Treasury Bank ATM

Withdrawal Successful

Savings Balance

Thank You For Banking With Us

Contact Us

This is a wireframe of a Treasury Bank ATM withdrawal success screen. It features a white rounded rectangle centered on a light gray background. The screen displays the title 'Treasury Bank ATM', followed by the status 'Withdrawal Successful' and a 'Savings Balance' input field. Below this is a 'Thank You For Banking With Us' message and a 'Contact Us' link at the bottom.



