



>>>> Users functions

```
// Registers a new user in the system
public void registerUser(User userDetails);

// Fetches user details based on user ID
public User getUserById(Long userId);

// Updates user profile information
public void updateUserProfile(Long userId, User updatedUserDetails);

// Fetches user profile data
public UserProfile getUserProfile(Long userId);

// Authenticates a user and returns a JWT token
public String authenticateUser(String email, String password);

// Fetches all users in the system (admin function)
public List<User> getAllUsers();

// Deletes a user account (admin function)
public void deleteUser(Long userId);

// Checks if a user exists based on email
public boolean userExists(String email);
```

>>>>> Loan Functions

```
// Applies for a loan on behalf of a user
public void applyForLoan(Long userId, Loan loanDetails);

// Fetches loan details based on the loan ID
public Loan getLoanById(Long loanId);

// Fetches all loans taken by a specific user
public List<Loan> getAllLoansByUserId(Long userId);

// Fetches the current status (Pending, Approved, Rejected) of a specific loan
public LoanStatus getLoanStatus(Long loanId);

// Approves or rejects a loan application
public void updateLoanStatus(Long loanId, LoanStatus status);

// Calculates monthly payment for a loan based on amount, interest rate, and tenure
public BigDecimal calculateMonthlyPayment(Long loanId);

// Fetches all loans currently pending approval
public List<Loan> getAllPendingLoans();

// Counts the number of payments made by a user for a specific loan
public int countPaymentsByLoanId(Long loanId, Long userId);

// Creates a loan with zero downpayment for a user
public Loan createLoanWithZeroDownpayment(Long userId, Loan loanDetails);

// Creates a transaction when the user makes the initial downpayment for a loan
public Transaction createDownpaymentTransaction(Long loanId, BigDecimal downpaymentAmount);

// Fetches all loans for admin view
public List<Loan> getAllLoans(); // Admin function to fetch all loans
```

>>>Transaction function

```
// Makes a loan payment and updates the corresponding transaction
public void makeLoanPayment(Long loanId, BigDecimal paymentAmount);

// Fetches all transactions related to a specific loan
public List<Transaction> getAllTransactionsByLoanId(Long loanId);

// Fetches all transactions made by a user across all their loans
public List<Transaction> getAllTransactionsByUser(Long userId);

// Fetches the most recent transaction for a specific loan
public Transaction getRecentTransactionByLoanId(Long loanId);

// Calculates the remaining balance for a specific loan
public BigDecimal calculateRemainingBalance(Long loanId);

// Fetches a specific transaction by transaction ID
public Transaction getTransactionById(Long transactionId);
```