```java
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Scanner;

class Book {
    private String title;
    private String author;
    private String isbn;
    private boolean isBorrowed;

    // Constructor
    public Book(String title, String author, String isbn) {
        this.title = title;
        this.author = author;
        this.isbn = isbn;
        this.isBorrowed = false;
    }

    // Getters and Setters
    public String getTitle() {
        return title;
    }

    public String getAuthor() {
        return author;
    }

    public String getIsbn() {
        return isbn;
    }

    public boolean isBorrowed() {
        return isBorrowed;
    }

    public void borrowBook() {
        this.isBorrowed = true;
    }

    public void returnBook() {
        this.isBorrowed = false;
    }

    @Override
    public String toString() {
        return "Book{" +
                "Title='" + title + '\'' +
                ", Author='" + author + '\'' +
                ", ISBN='" + isbn + '\'' +
                ", Status=" + (isBorrowed ? "Borrowed" : "Available") +
                '}';
    }
}

public class LibraryManagementSystem {
    private static ArrayList<Book> books = new ArrayList<>();
    private static Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {
        boolean running = true;

        while (running) {
            System.out.println("\nLibrary Management System");
            System.out.println("1. Add Book");
            System.out.println("2. Update Book");
            System.out.println("3. Delete Book");
            System.out.println("4. Search Book");
            System.out.println("5. Borrow Book");
            System.out.println("6. Return Book");
            System.out.println("7. Display All Books");
            System.out.println("8. Exit");
            System.out.print("Enter your choice: ");

            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline

            switch (choice) {
                case 1:
                    addBook();
                    break;
                case 2:
                    updateBook();
                    break;
                case 3:
                    deleteBook();
                    break;
                case 4:
                    searchBook();
                    break;
                case 5:
                    borrowBook();
                    break;
                case 6:
                    returnBook();
                    break;
                case 7:
                    displayAllBooks();
                    break;
                case 8:
                    running = false;
                    System.out.println("Exiting Library Management System.");
                    break;
                default:
                    System.out.println("Invalid choice. Please try again.");
            }
        }
    }

    private static void addBook() {
        System.out.print("Enter book title: ");
        String title = scanner.nextLine();
        System.out.print("Enter book author: ");
        String author = scanner.nextLine();
        System.out.print("Enter book ISBN: ");
        String isbn = scanner.nextLine();

        books.add(new Book(title, author, isbn));
        System.out.println("Book added successfully.");
    }

    private static void updateBook() {
        System.out.print("Enter ISBN of the book to update: ");
        String isbn = scanner.nextLine();
        for (Book book : books) {
            if (book.getIsbn().equals(isbn)) {
                System.out.print("Enter new title: ");
                book = new Book(scanner.nextLine(), book.getAuthor(),
book.getIsbn()); // Simple replacement
                System.out.println("Book updated successfully.");
                return;
            }
        }
        System.out.println("Book not found.");
    }

    private static void deleteBook() {
        System.out.print("Enter ISBN of the book to delete: ");
        String isbn = scanner.nextLine();
        for (Book book : books) {
            if (book.getIsbn().equals(isbn)) {
                books.remove(book);
                System.out.println("Book deleted successfully.");
                return;
            }
        }
        System.out.println("Book not found.");
    }

    private static void searchBook() {
        System.out.print("Enter title or author or ISBN to search: ");
        String query = scanner.nextLine().toLowerCase();
        for (Book book : books) {
            if (book.getTitle().toLowerCase().contains(query) ||
                book.getAuthor().toLowerCase().contains(query) ||
                book.getIsbn().equals(query)) {
                System.out.println(book);
                return;
            }
        }
        System.out.println("Book not found.");
    }

    private static void borrowBook() {
        System.out.print("Enter ISBN of the book to borrow: ");
        String isbn = scanner.nextLine();
        for (Book book : books) {
            if (book.getIsbn().equals(isbn)) {
                if (!book.isBorrowed()) {
                    book.borrowBook();
                    System.out.println("Book borrowed successfully.");
                } else {
                    System.out.println("Book is already borrowed.");
                }
                return;
            }
        }
        System.out.println("Book not found.");
    }

    private static void returnBook() {
        System.out.print("Enter ISBN of the book to return: ");
        String isbn = scanner.nextLine();
        for (Book book : books) {
            if (book.getIsbn().equals(isbn)) {
                if (book.isBorrowed()) {
                    book.returnBook();
                    System.out.println("Book returned successfully.");
                } else {
                    System.out.println("Book was not borrowed.");
                }
                return;
            }
        }
        System.out.println("Book not found.");
    }

    private static void displayAllBooks() {
        if (books.isEmpty()) {
            System.out.println("No books available.");
        } else {
            for (Book book : books) {
                System.out.println(book);
            }
        }
    }
}
```