```python
# List to store student records (acting as a simple in-memory database)
students = []

def add_student(name, roll_number, grade):
    """Adds a new student record to the system."""
    student = {"name": name, "roll_number": roll_number, "grade": grade}
    students.append(student)
    print(f"Student {name} added successfully.")

def view_students():
    """Displays all student records in the system."""
    if not students:
        print("No student records available.")
        return

    print("\n--- Student Records ---")
    for student in students:
        print(f"Name: {student['name']}, Roll Number:
{student['roll_number']}, Grade: {student['grade']}")
    print("----------------------\n")

def search_student(roll_number):
    """Searches for a student by their roll number."""
    for student in students:
        if student['roll_number'] == roll_number:
            print(f"\nStudent Found:")
            print(f"Name: {student['name']}, Roll Number:
{student['roll_number']}, Grade: {student['grade']}")
            return student
    print(f"Student with Roll Number {roll_number} not found.")
    return None

def update_student(roll_number, new_name=None, new_grade=None):
    """Updates an existing student's name or grade."""
    for student in students:
        if student['roll_number'] == roll_number:
            if new_name:
                student['name'] = new_name
            if new_grade:
                student['grade'] = new_grade
            print(f"Student with Roll Number {roll_number} updated
successfully.")
            return True
    print(f"Student with Roll Number {roll_number} not found.")
    return False

def delete_student(roll_number):
    """Deletes a student record from the system."""
    global students
    initial_len = len(students)
    students = [student for student in students if student['roll_number'] !=
roll_number]
    if len(students) < initial_len:
        print(f"Student with Roll Number {roll_number} deleted
successfully.")
        return True
    print(f"Student with Roll Number {roll_number} not found.")
    return False
```

```python
def display_menu():
    """Displays the main menu of the Student Record Management System."""
    print("\n--- Student Record Management System ---")
    print("1. Add Student")
    print("2. View All Students")
    print("3. Search Student by Roll Number")
    print("4. Update Student Record")
    print("5. Delete Student Record")
    print("6. Exit")
    print("--------------------------------------")

def main():
    """Main function to run the Student Record Management System."""
    while True:
        display_menu()
        choice = input("Enter your choice: ")

        if choice == '1':
            name = input("Enter student name: ")
            roll_number = input("Enter student roll number: ")
            grade = input("Enter student grade: ")
            add_student(name, roll_number, grade)
        elif choice == '2':
            view_students()
        elif choice == '3':
            roll_number = input("Enter roll number to search: ")
            search_student(roll_number)
        elif choice == '4':
            roll_number = input("Enter roll number of student to update: ")
            new_name = input("Enter new name (leave blank to keep current): ")
            new_grade = input("Enter new grade (leave blank to keep current): ")
            update_student(roll_number, new_name if new_name else None, new_grade if new_grade else None)
        elif choice == '5':
            roll_number = input("Enter roll number of student to delete: ")
            delete_student(roll_number)
        elif choice == '6':
            print("Exiting Student Record Management System. Goodbye!")
            break
        else:
            print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```