DIAGRAMS

1. Flowchart: User Registration & Login Process

Diagram Overview:

This flowchart represents the process of **user registration and login** in the application. It covers both the registration and login steps, including validation, error handling, and successful login.

Steps:

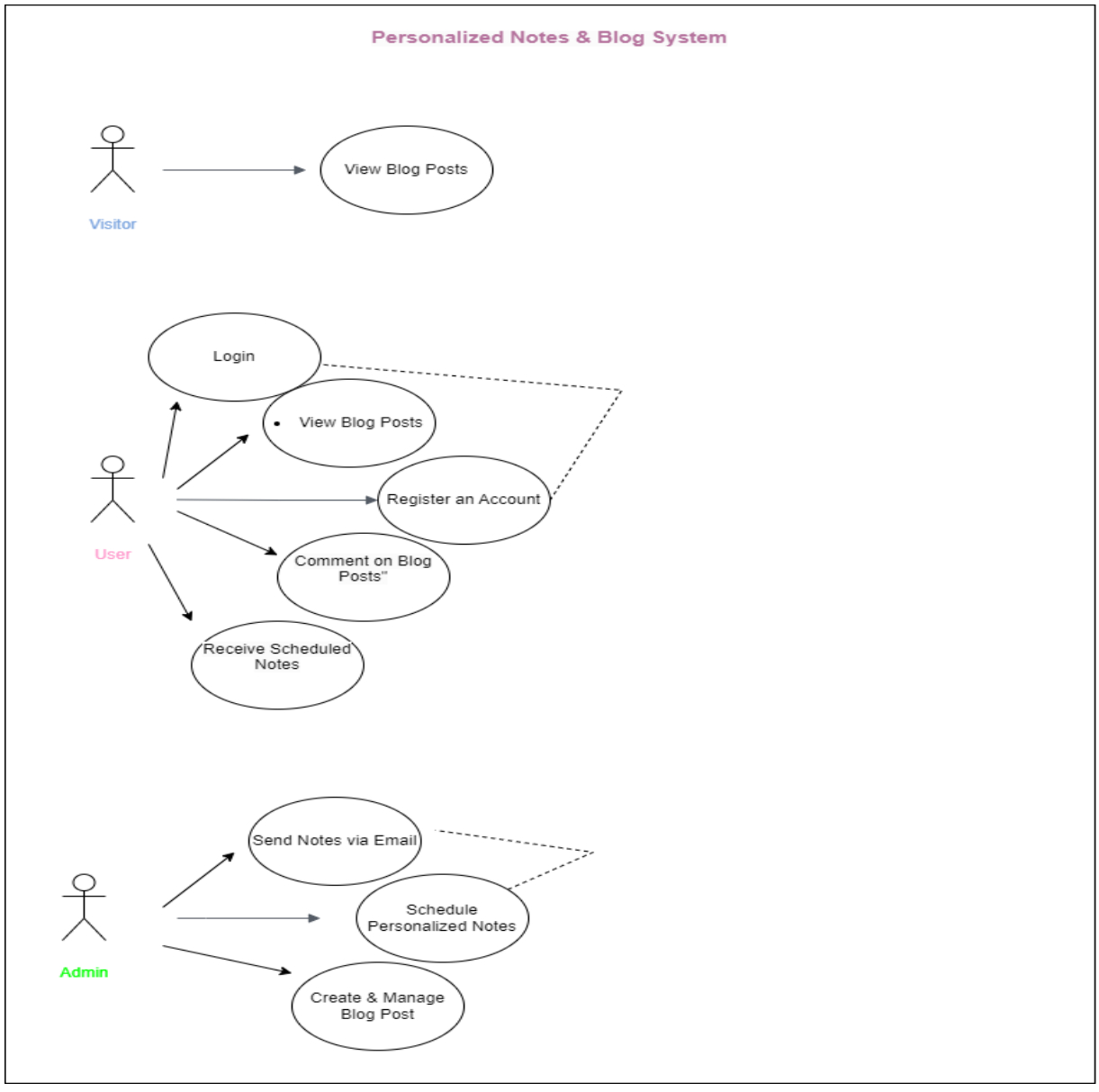
1. **Start**:
   - This marks the beginning of the flowchart.
2. **User enters registration info**:
   - The user inputs their **name**, **email**, and **password** in the registration form. This is the starting point of the registration process.
3. **Validate input**:
   - The system validates the input to check if the email is in the correct format, the password meets security requirements, and whether the email already exists in the database.
4. **Is input valid?** (Decision):
   - The system checks if the user input is valid.
   - If valid, the process moves to creating the account.
   - If not, the system displays an error message.
5. **Show error message** (If input is invalid):
   - If the input is invalid (e.g., the email is incorrectly formatted or the password is too weak), the system shows an error message prompting the user to correct the input.
6. **Create account** (If input is valid):
   - If the input is valid, the system creates an account in the database for the user.
7. **Redirect to login page**:
   - After the account is created, the user is redirected to the login page to access their newly created account.
8. **User enters login info**:
   - The user inputs their **email** and **password** to log in to the system.
9. **Check credentials**:
   - The system checks if the entered email and password match the records in the database.
10. **Do credentials match?** (Decision):
    - If the credentials match, the user is logged in successfully.
    - If not, an "invalid login" message is shown.
11. **Show invalid login message** (If credentials don't match):
    - If the login credentials are incorrect, an error message is displayed, and the user is prompted to try again.
12. **Redirect to dashboard/home page** (If credentials match):

- If the credentials are correct, the user is redirected to their dashboard or home page.

13. **End**:
    - The process ends after a successful login.

Diagram Flow:

- **Start → User enters registration info → Validate input → Is input valid? → Yes → Create account → Redirect to login page → User enters login info → Check credentials → Do credentials match? → Yes → Redirect to dashboard/home page → End**
  → No (Invalid Input) → Show error message → Go back to registration info (loop)
  → No (Invalid Login) → Show invalid login message → Go back to login info (loop)



---

2. Use Case Diagram: User Roles and System Interactions

Diagram Overview:

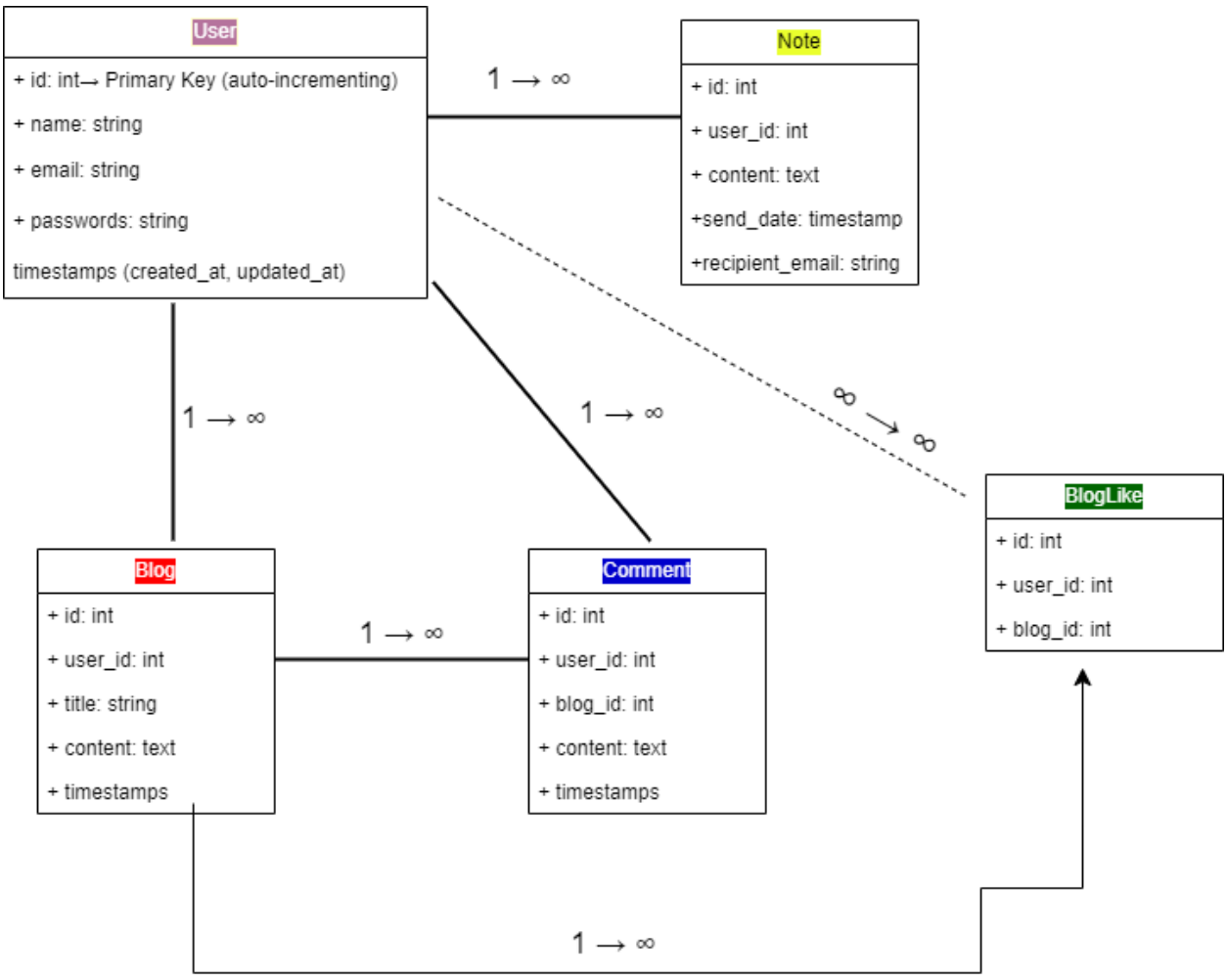
This use case diagram shows the **different user roles** in the system (Administrator and User) and their interactions with the system. It also outlines the **key features** available to each user.

Steps:

1. **User (Actor)**:
    - Represents a **regular user** who can interact with the application by registering, logging in, viewing the blog, and commenting.
2. **Administrator (Actor)**:
    - Represents the **administrator** who has additional privileges, such as managing user accounts, creating personalized notes, and moderating the blog.
3. **Register Account (Use Case)**:
    - Both the **user** and **administrator** can register an account, where the user submits registration details.
4. **Login (Use Case)**:
    - Both the **user** and **administrator** can log in to their accounts.
5. **View Blog (Use Case)**:
    - Both users and visitors can view the blog posts, but only the administrator can modify them.
6. **Comment on Blog (Use Case)**:
    - Users can comment on blog posts. Administrators can also comment, but they also have the ability to manage posts.
7. **Create Personalized Notes (Use Case)**:
    - The **administrator** can create personalized notes to be sent to users based on their email addresses.
8. **Delete Blog Post (Use Case)**:
    - Only the **administrator** can delete or modify blog posts.
9. **End**:
    - This represents the end of the system's interactions with the actors.

Diagram Flow:

- **Administrator** can interact with: **Register Account**, **Login**, **Create Personalized Notes**, **Delete Blog Post**, **View Blog**, **Comment on Blog**.
- **User** can interact with: **Register Account**, **Login**, **View Blog**, **Comment on Blog**.
- **Visitor** can only interact with: **View Blog**.



---

3. Sequence Diagram: User Registration Process

Diagram Overview:

The sequence diagram illustrates the **detailed steps** involved in the **user registration process**, showing the interaction between the user, the system, and the database.

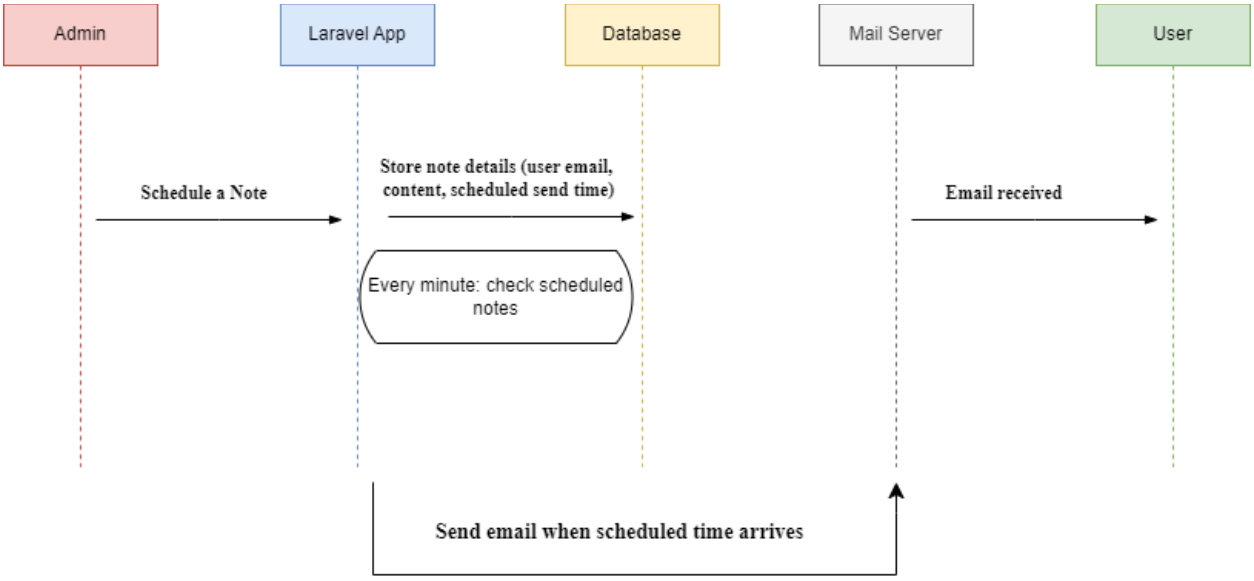
Steps:

1. **User** sends a registration request to the **system** with their personal details (name, email, password).
2. The **system** validates the input to ensure that the email is correctly formatted and that the password meets security standards.
3. If the input is valid, the **system** communicates with the **database** to store the user's information.

4. Once the account is created, the **system** sends a confirmation response back to the **user**, and the user is redirected to the login page.

Diagram Flow:

- **User → System** (sends registration info).
- **System → Database** (stores user data).
- **System → User** (confirmation and redirect to login).



---

4. Activity Diagram: Personalized Notes Creation

Diagram Overview:

This activity diagram represents the process for creating **personalized notes** by the **administrator**. It covers all the activities involved in creating and scheduling personalized notes to users.

Steps:

1. **Administrator** starts the process of creating a personalized note.
2. The **administrator** enters the **email** of the user to receive the note.
3. The **administrator** sets a **specific date** for when the note will be sent.
4. The **system** stores the note with the set date and user email.
5. The **system** schedules the note to be sent on the specified date.
6. Once the note is sent, the **system** confirms the successful delivery.

7. The process ends once the note is sent and confirmed.

Diagram Flow:

- **Administrator → Enter user email → Set sending date → System stores and schedules note → System confirms sending → End**