

Project Elevator pitch:

Marvel Nexus is a web application for Marvel comic enthusiasts to explore characters and manage their comic collections. It offers detailed character profiles, a comic library with personalized recommendations, and a user-friendly interface for browsing and tracking favorites. All you need in one application for any comic fan.

Personas:

Persona 1: Luke

Name: Luke

Gender: Male

Age: 20

Occupation: Full-time college student

About:

Luke is a Marvel fan who enjoys both the comics and movies. He loves diving into the backstories of characters and often uses this knowledge to impress friends. As a student, his schedule is packed with classes, assignments, and extracurricular activities, leaving little time for hobbies. He uses his downtime to explore Marvel lore and unwind.

Goals:

- Discover lesser-known Marvel characters and their story arcs.
- Keep track of comics he wants to read during breaks.
- Share interesting Marvel facts with friends.

Needs:

- A user-friendly app to quickly search for characters and comics.
- A way to save favorite characters and comics to revisit later.
- Personalized recommendations to explore new comics during free time.

Frustrations:

- Feels overwhelmed by the vast amount of Marvel content available.
- Finds it hard to remember which comics he has already read or wants to read.
- Wishes there was a single platform to explore Marvel characters and comics without needing multiple apps or websites.

Persona 2: Sam

Name: Sam

Gender: Male

Age: 40

Occupation: Accountant

About:

Sam is a lifelong Marvel fan and an avid comic book collector. His collection includes both physical and digital comics, ranging from classic issues to modern storylines. He takes pride in organizing and cataloging his collection but often struggles to keep track of what he owns and what he still needs to complete certain series. Sam enjoys discovering related series and characters to expand his collection.

Goals:

- Organize and catalog his comic collection efficiently.
- Discover new comics and series that align with his interests.
- Complete missing issues in his favorite story arcs or series.

Needs:

- A robust app to track his physical and digital comic inventory.
- Recommendations for comics related to his existing collection.
- A search feature to quickly find details about specific comics or series.

Frustrations:

- Finds it challenging to keep track of comics he owns versus ones he still needs.
- Struggles to discover related series without manually searching through extensive lists.
- Wishes there was a tool to help him visualize and manage his collection more effectively.

Use Cases:

1. Explore Characters

- **Actor:** User
- **Goal:** Search for characters and view their details.
- **Steps:**
 1. User enters a search query (e.g., character name).
 2. System fetches matching data from the Marvel API.
 3. System displays a list of characters matching the query.
 4. User selects a character from the list.

5. System fetches and displays the character's detailed profile, including name, description, image, and affiliations.

2. Explore Comics

6. **Actor:** User
7. **Goal:** Search for comics and view their details.
8. **Steps:**
 - User enters a search query (e.g., comic title or issue number).
 - System fetches matching data from the Marvel API.
 - System displays a list of comics matching the query.
 - User selects a comic from the list.
 - System fetches and displays the comic's detailed profile, including title, issue number, creators, and release date.

3. Save Comics

- **Actor:** User
- **Goal:** Add a comic to the favorites list.
- **Steps:**
 1. User browses or searches for comics.
 2. User selects a comic to save.
 3. System saves the comic to the user's database record.
 4. User views the comic in their "Favorites" section.

4. Save Characters

- **Actor:** User
- **Goal:** Add a character to the favorites list.
- **Steps:**
 1. User browses or searches for characters.
 2. User selects a character to save.
 3. System saves the character to the user's database record.
 4. User views the character in their "Favorites" section.

5. Create User

- **Actor:** User
- **Goal:** Register a new account.
- **Steps:**
 1. User navigates to the registration page.
 2. User enters required details (name, email, password).
 3. System validates the input and hashes the password.

4. System saves the user details in the database.
5. User receives a confirmation message and can log in.

6. Log In

- **Actor:** User
- **Goal:** Authenticate and access the application.
- **Steps:**
 1. User navigates to the login page.
 2. User enters email and password.
 3. System validates credentials using OAuth2.
 4. User is authenticated and redirected to the dashboard.

7. Create Collection

- **Actor:** User
- **Goal:** Create a new collection for comics or characters.
- **Steps:**
 1. User navigates to the "Collections" section.
 2. User clicks on "Create Collection."
 3. User enters a collection name and optional description.
 4. System saves the collection to the database.
 5. User sees the new collection in their list.

8. Add to Collection

- **Actor:** User
- **Goal:** Add a comic or character to a specific collection.
- **Steps:**
 1. User selects a comic or character.
 2. User chooses a collection to add the item to.
 3. System updates the collection in the database.
 4. User views the updated collection.

9. View Collections

- **Actor:** User
- **Goal:** Browse existing collections and their contents.
- **Steps:**
 1. User navigates to the "Collections" section.
 2. System fetches and displays all user collections.
 3. User selects a collection to view its contents.
 4. System displays the items (comics or characters) in the collection.

10. Remove from Favorites or Collections

- **Actor:** User
- **Goal:** Remove a comic or character from favorites or a collection.
- **Steps:**
 1. User navigates to the "Favorites" or "Collections" section.
 2. User selects an item to remove.
 3. System removes the item from the database.
 4. User no longer sees the item in the list.

11. Update User Account

- **Actor:** User
- **Goal:** Update account settings like email, password, etc.
- **Steps:**
 1. User navigates to the "Settings" page.
 2. User updates account information.
 3. System saves information in the database.
 4. User sees the updated information applied.

12. Log Out

- **Actor:** User
- **Goal:** End the session and log out.
- **Steps:**
 1. User clicks the "Log Out" button.
 2. System invalidates the user session.
 3. User is redirected to the login page.