



Unleashing the Power of GitHub: A Journey to Collaboration and Innovation

Introduction to GitHub

Proposed Solution

Collaboration on GitHub

Open Source on GitHub

GitHub in Education

Functional requirements

Non-Functional requirements

Modelling and Diagrams

Conclusion



Introduction to GitHub

GitHub is a web-based platform that allows users to store and manage their code repositories. It was founded in 2008 by Tom Preston-Werner, Chris Wanstrath, and PJ Hyett.

GitHub has become one of the most popular platforms for developers to collaborate on projects and share code with others. It provides an easy-to-use interface for managing code, tracking changes, and collaborating with others.



Proposed Solution

The power of Github lies in its ability to facilitate collaboration and innovation among developers. With its user-friendly interface, Github allows developers to easily share their code, track changes, and collaborate with others on projects. However, to fully unleash the power of Github, it is important to have a clear understanding of how to use its features effectively.

One way to do this is by creating a project board to manage tasks and issues. This allows team members to see what needs to be done, who is responsible for each task, and the status of each issue. Additionally, using Github's pull request feature can help streamline the code review process and ensure that all changes are thoroughly reviewed before being merged into the main branch.



Collaboration on GitHub

One of the biggest advantages of using Github is the ability to collaborate with other developers. Users can easily share their code with others, and multiple people can work on the same project simultaneously.

Github also provides tools for reviewing and commenting on code changes, making it easier for developers to provide feedback and suggest improvements to each other's work.



Open Source on GitHub

Github has become one of the most popular platforms for open source development. Open source projects are projects that are freely available for anyone to use, modify, and distribute.

Github provides a platform for developers to share their open source projects with the world, and allows others to contribute to these projects by submitting changes and improvements.



GitHub in Education

GitHub is also becoming increasingly popular in education. Many universities and schools are using GitHub as a tool for teaching programming and software development.

GitHub provides a way for students to collaborate on projects and learn from each other, and also allows teachers to easily review and grade assignments.





Functional requirements

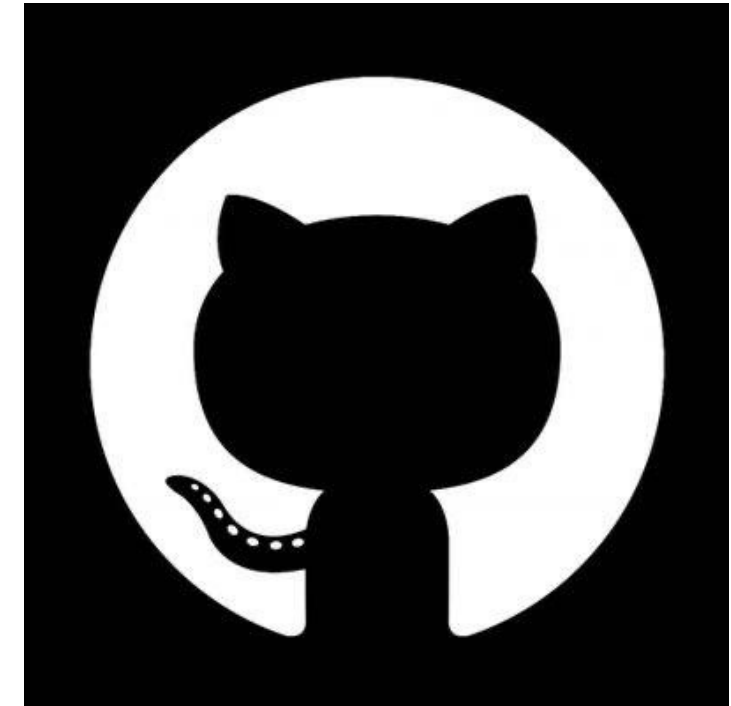
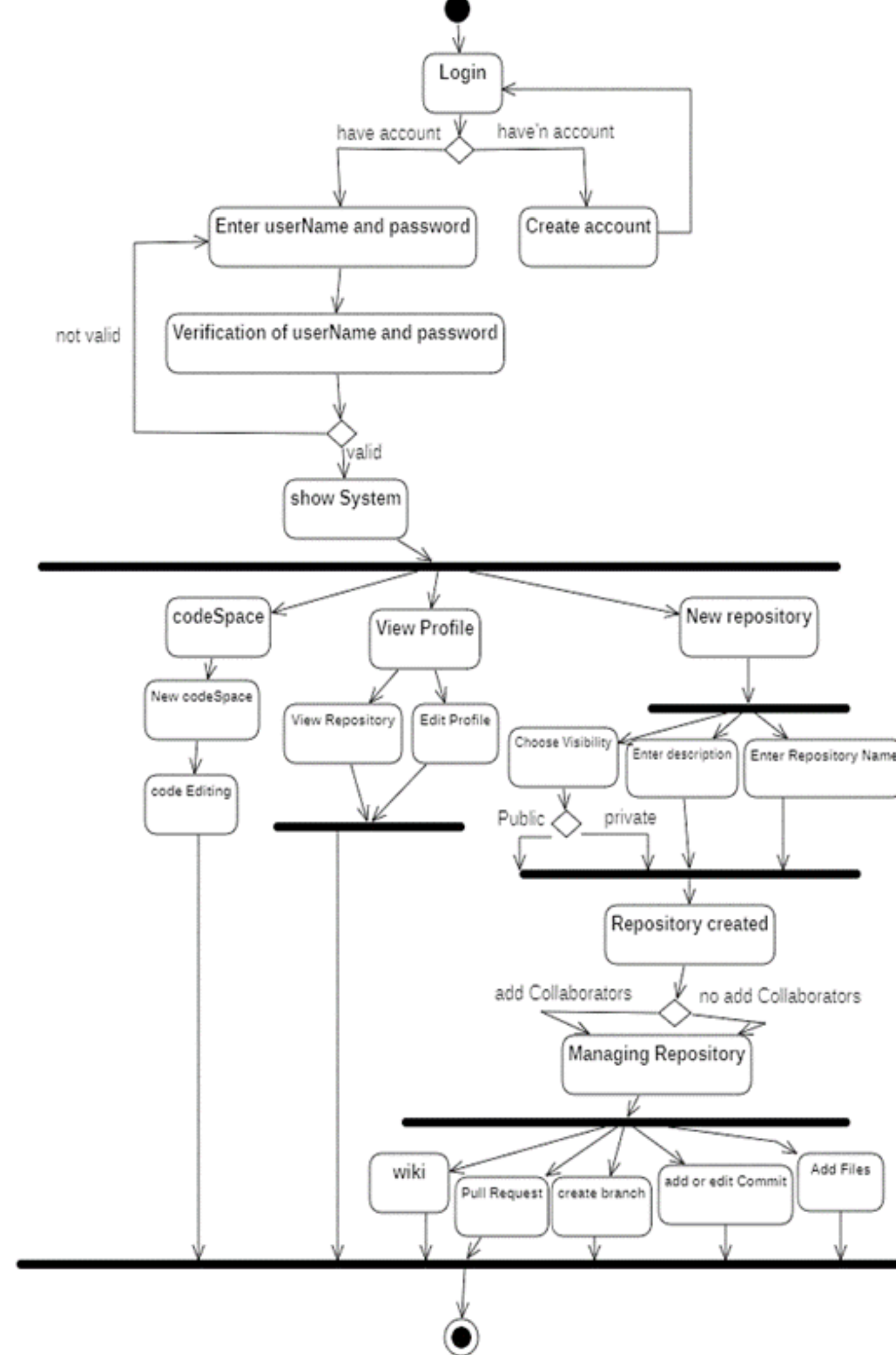
Functional	Description	Functional	Description
User Registration	The system should allow users to create an account by providing information such as username, email address, and password.	Search	The system allows the user to a search box may appear for him and he will type about which he wants to obtain information.
Login	The system allows for users to log into their account by entering their email and password.	Create and manage repositories	Users should be able to create and manage repositories, which are collections of code files.



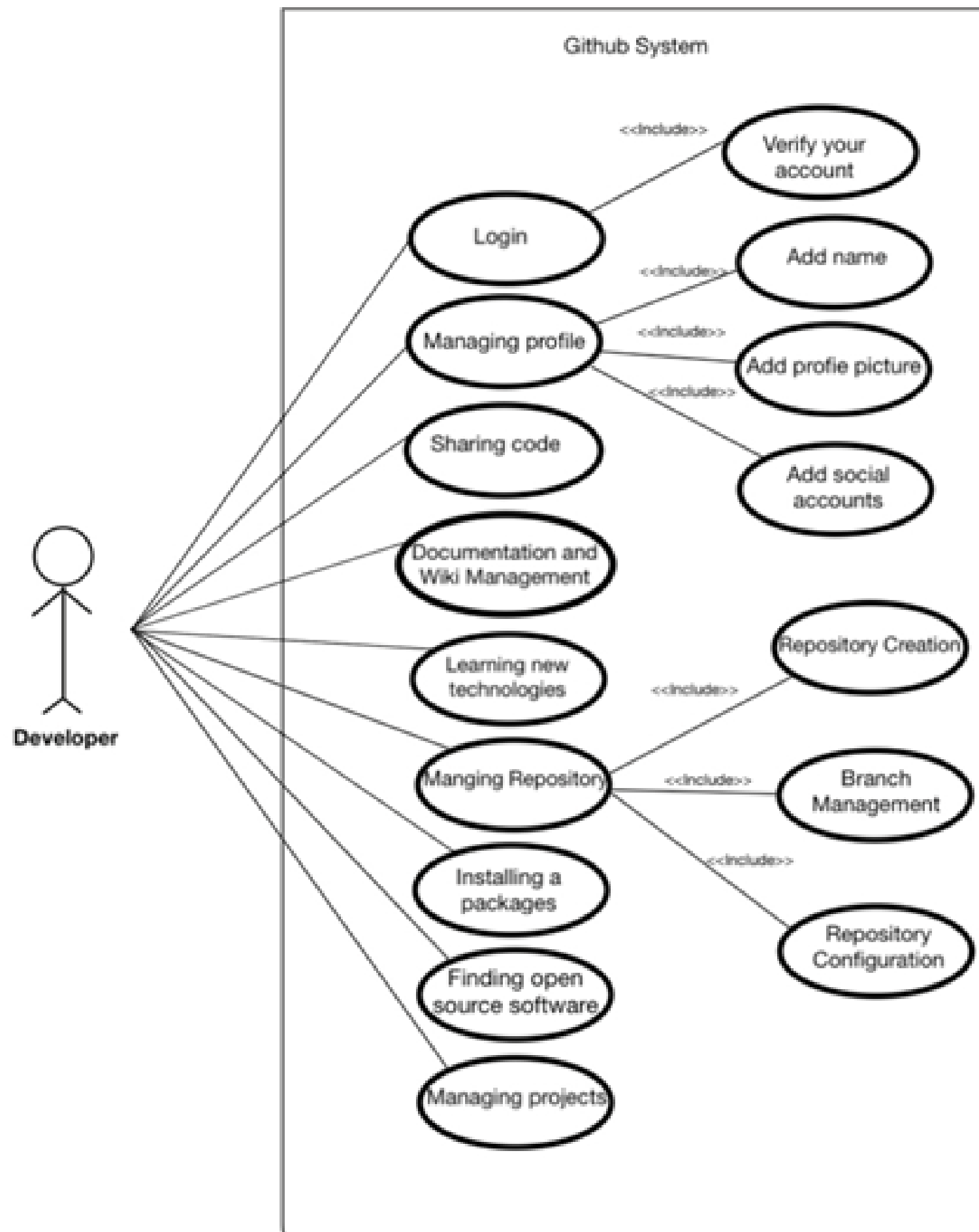
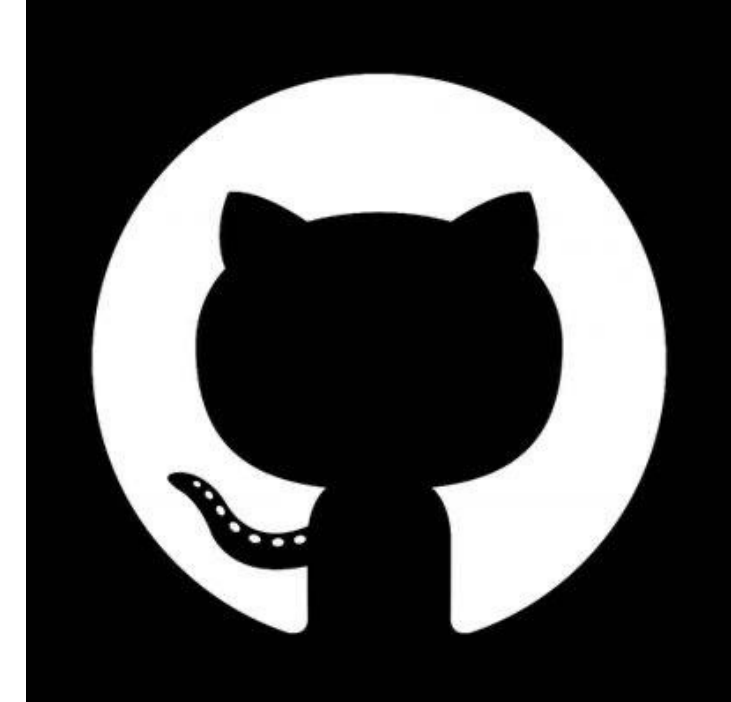
Non-Functional requirements

Non-Functional	Description	Non-Functional	Description
Performance	The system should be responsive and scalable to handle a large number of users and repositories.	Ease of use	The system should be easy to use and understand.
Availability	The system must available 24/7.	Securitys	The system should be secure and protect users' data from unauthorized access.

1. Activity diagram:



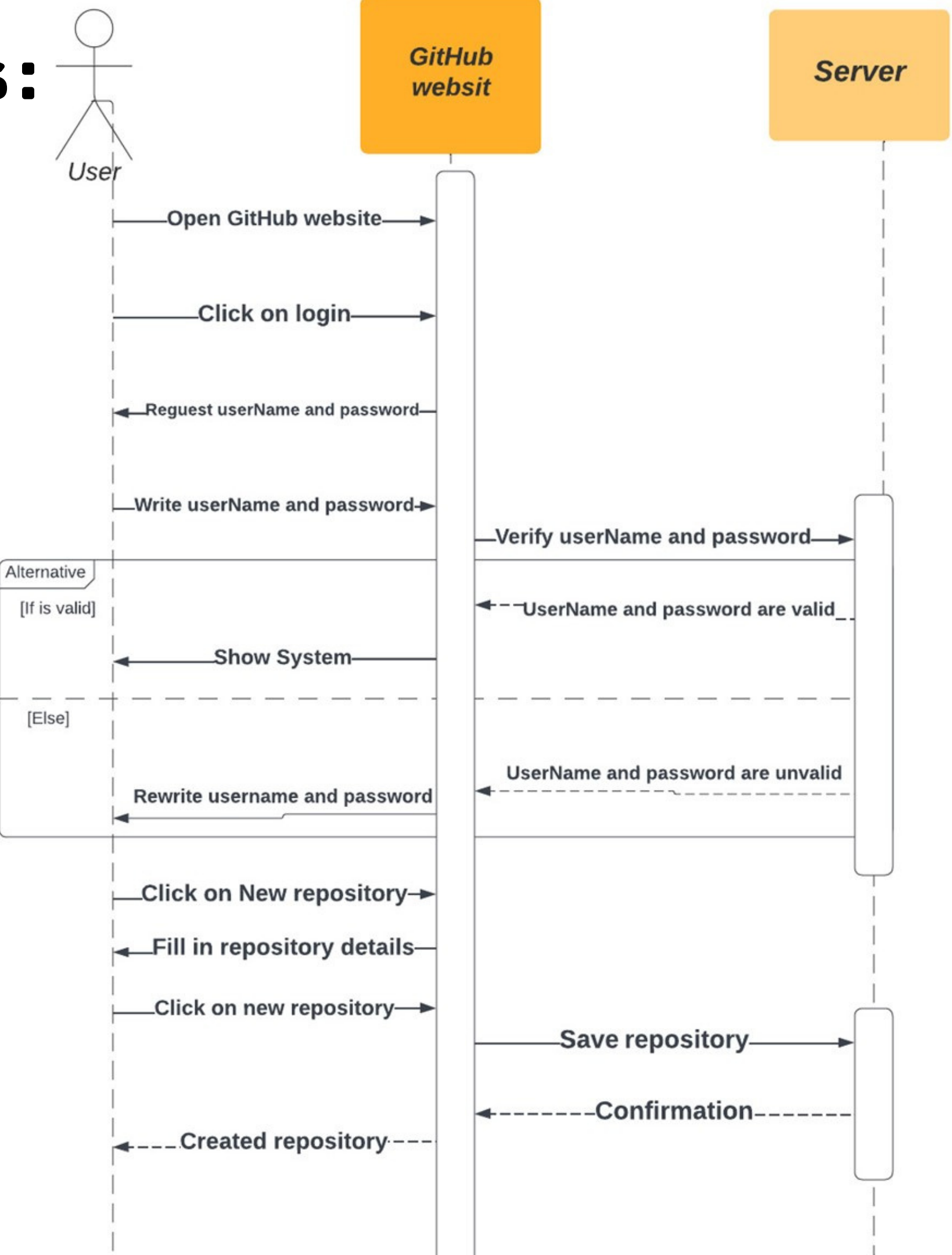
2. Project Use Case Modeling:



Developer case: Managing Repository
Actor : Developer
Description : Repository management includes supervision and control of the development and maintenance of the hosted code base, and includes tasks such as creating the Repository , organizing files and folders, tracking changes, cooperating with others, and ensuring the stability and progress of the project. In general, Repository management allows for effective cooperation, efficient code management and streamlining of development processes.
Data : various types of data are utilized. Here are some of the key data elements involved: Repository metadata, Source code files and Commit history , Branches and tags , Collaborator data Issues and pull requests, etc
Simulus : The stimulus can be any action or event that triggers a response or operation in the GitHub repository management process. Some examples of stimuli in this context include: Creating a new repository ,Making changes to the source code files And Creating a new branch
Response : The response refers to the action or outcome that occurs as a result of the stimulus. the responses include various operations and behaviors, such as: Cloning the repository to the local machine, Staging and committing changes to the local repository ,Pushing changes to the remote repository.

Developer case: Managing profile
Actor : Developer
Description : Manage the profile and add the name, email and photo view and name of the company and social accounts .
Data : Account Information, Links and Profile Text .
Simulus : Managing the developer on his own account and providing a profile of himself .
Response : Account information modified and updated .

3. Creating Sequence Diagrams:



Conclusion

Github has become an essential tool for developers around the world. Its powerful features and easy-to-use interface make it an ideal platform for managing code, collaborating with others, and sharing open source projects.

Whether you're a professional developer or just starting out, Github is a valuable resource that can help you learn, grow, and succeed in the world of software development.





Thank you for your time!