

NewsLetterApp

Project of a native IOS application used to view the latest news.

Created by: Michał Zawadzki

Table of Contents

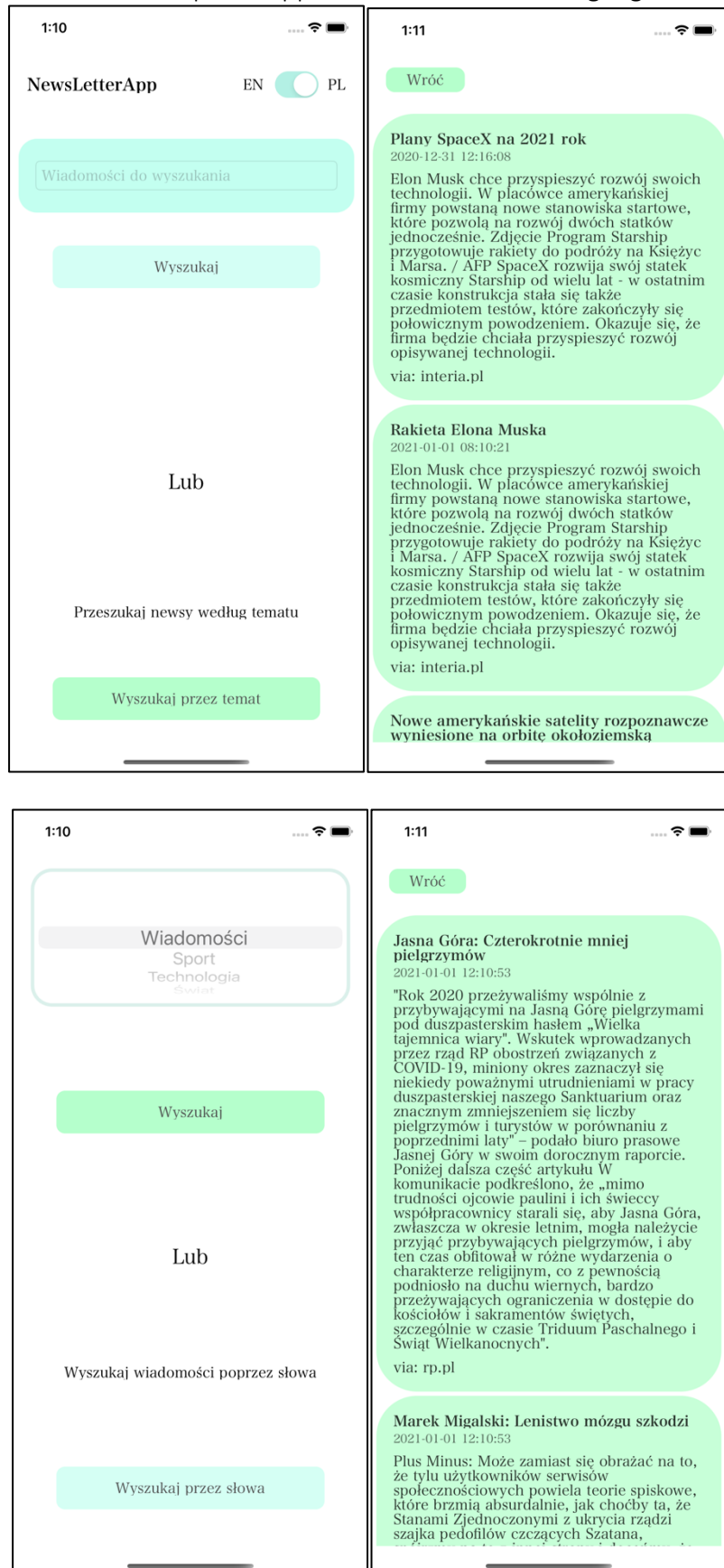
INTRODUCING	3
APPLICATION PRESENTATION	4
APPLICATION ARCHITECTURE	8
APPLICATION LAYERS AND COMMUNICATION BETWEEN THEM	9
USE CASE AND ACTIVITY DIAGRAMS	10

Introducing

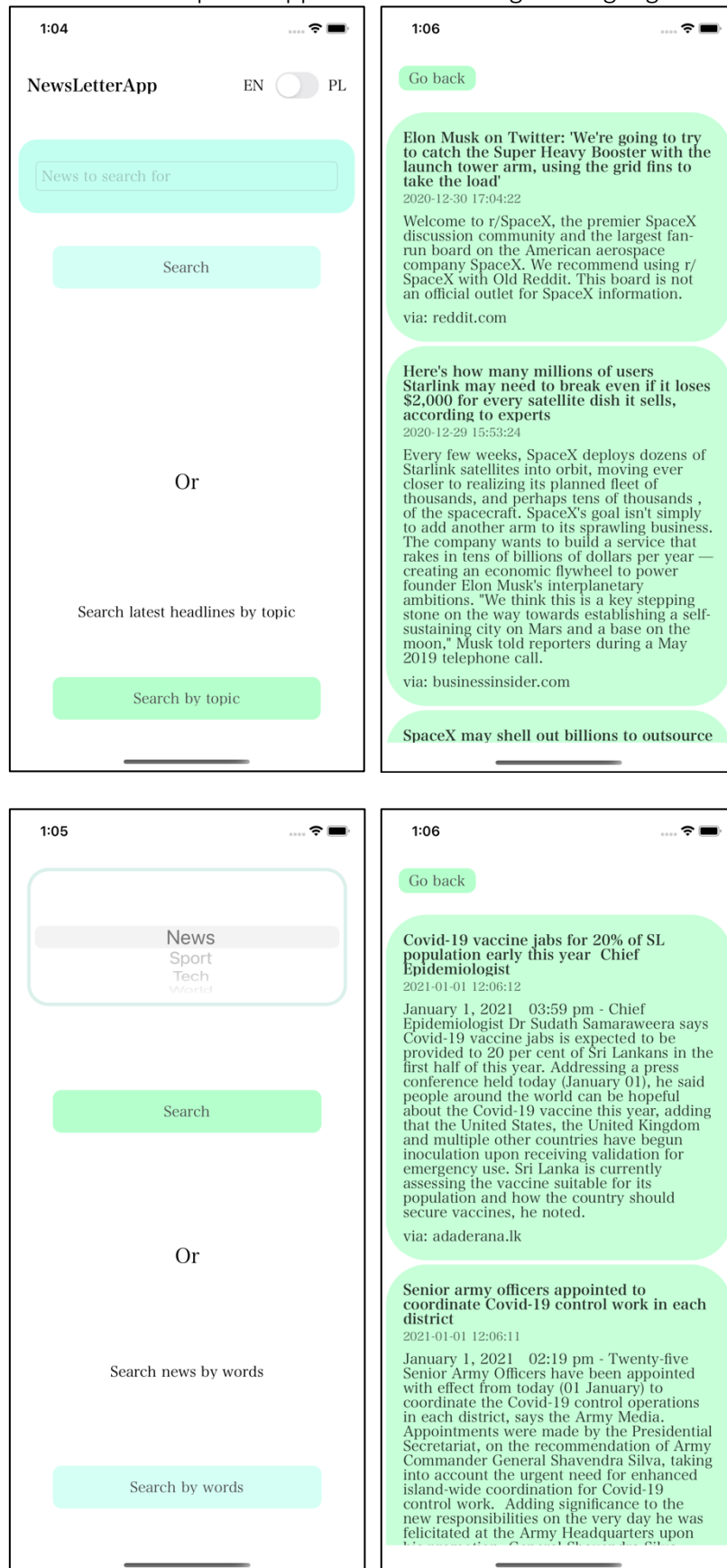
- **General information:** The application allows you to fetch from an external API (Newscatcher) and display 5 news about topic typed / selected by the user. The application supports 2 languages: Polish and English, after changing the language, the elements of the graphical user interface and the parameters of the request sent to the API change. In addition, a responsive layout of graphic elements has been used so that the application is adapted to both small and large devices. The application supports changing the color mode, adapting to the dark / light theme.
- **Functional requirements:**
 - The user can search for news of interest by entering his own phrase.
 - The user can search for news of interest by selecting one of the categories: News, Sport, Tech, World, Finance, Politics, Business, Economy, Entertainment.
 - The user can change the language of the application.
- **Nonfunctional requirements:**
 - Efficiency: By building a native application, it works immediately.
 - Usability: The application shows the date, title, summary and source of a post, thus providing a simple tool for viewing the news. In addition, for the user's convenience, the application adjusts to different device sizes and a dark / light theme.
 - Reliability: Application startup / shutdown / operation does not affect the operation of the system or other applications. In addition, the application has been protected against the possibility of returning incorrect data.
- **Requirements:**
 - Internet access
- **Glossary:**
 - Native application: An application created for a specific platform.
 - IOS: Operating system for mobile devices created by Apple.
 - API (Application Programming Interface): A collection of features within an application that provides the ability to interact with that application.
 - Server: A computer / program providing users with its resources / services.
 - View: Part of the application architecture responsible for data presentation.
 - Controller: Part of the application architecture responsible for application control and user interaction.
 - Model: The part of the application responsible for the storage and representation of data.

Application presentation

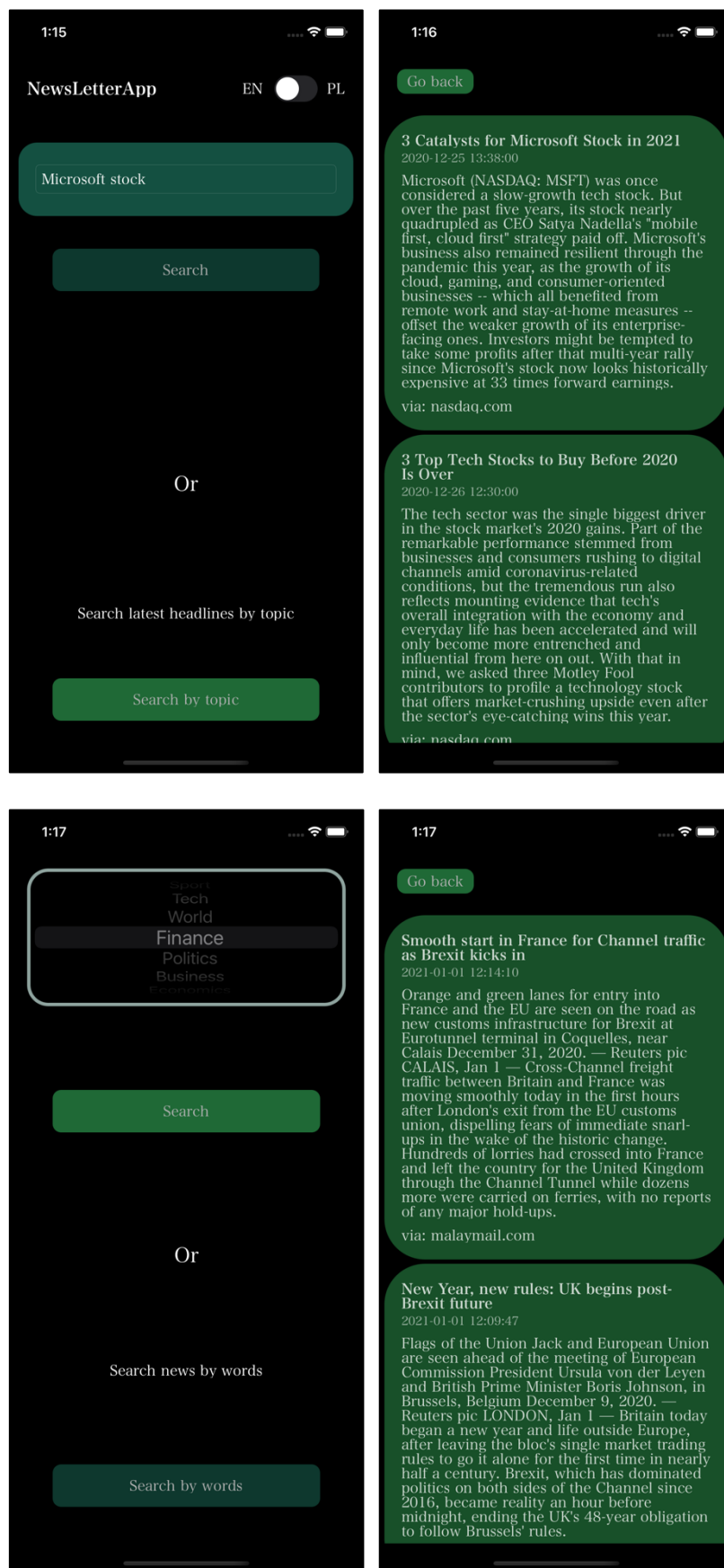
- Presentation on the example of Apple iPhone 11 - Polish language:



- Presentation on the example of Apple iPhone 11 - English language:



- Presentation on the example of Apple iPhone 11 with dark theme:



- Presentation on devices with different screen sizes:
 - iPod Touch (4-inch display):



- iPad Pro (11-inch display):

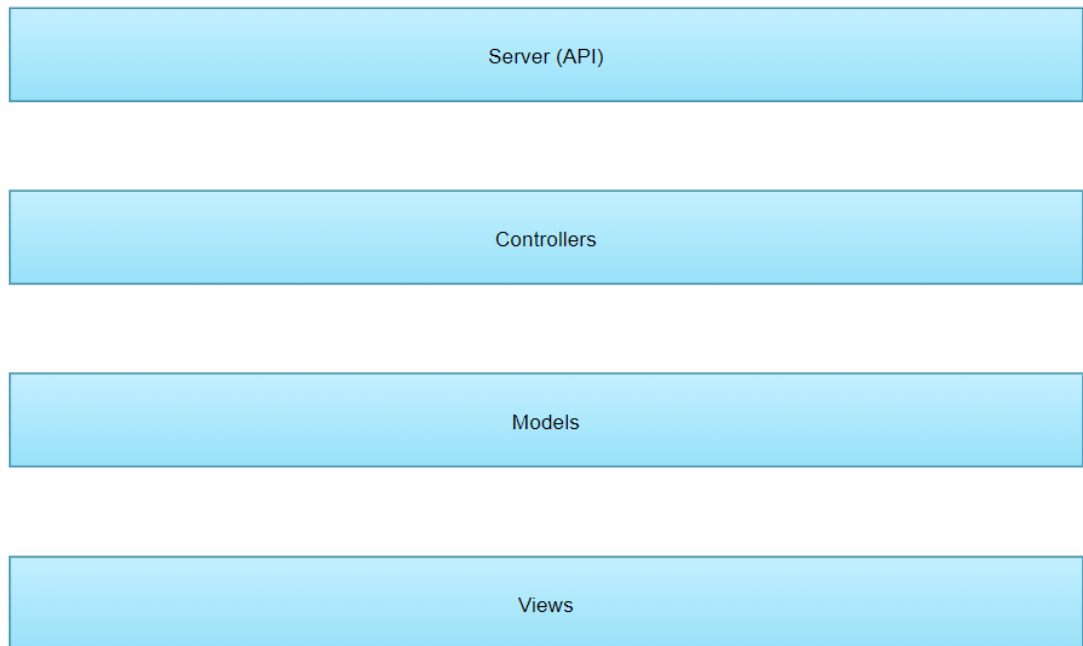


Application architecture

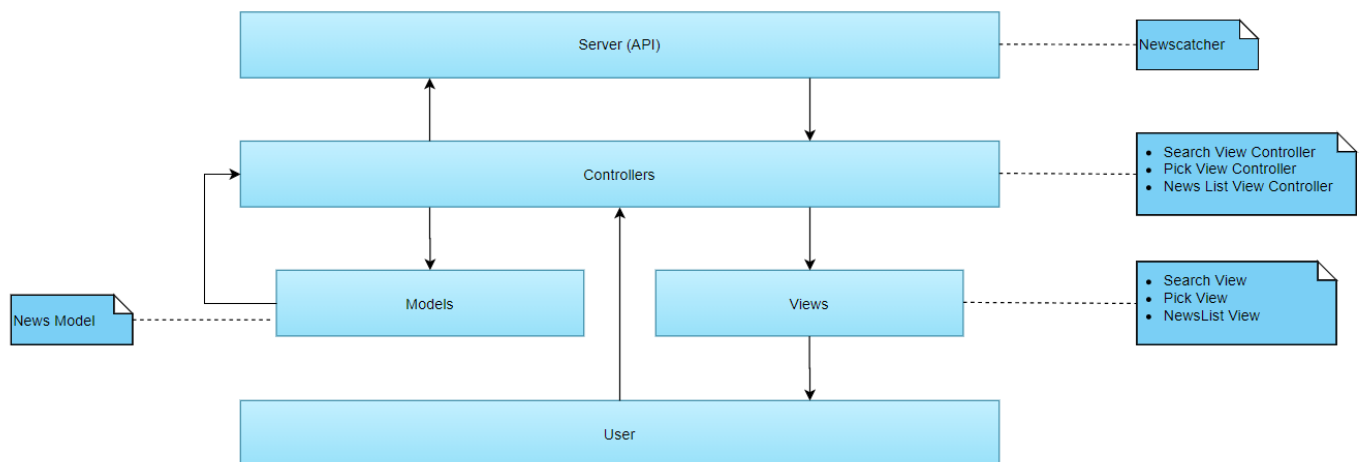
- The construction of the software architecture was based on two architectural patterns: MVC (Model - View - Controller) for building the application and Client - Server for communication of the application with an external API.
- Model:
 - NewsModel: A structure that maps data returned from the server.
- Views:
 - Search View: The initial view that allows you to search for news by phrase, change the language, and go to the view that allows you to search by category.
 - Pick View: Auxiliary view that allows you to search for news related to the selected category or return to Search View.
 - New List View: View that displays the messages downloaded from the server in the form of a scrolling list, it also allows you to return to the previous view depending on whether it was Search View or Pick View.
- Controllers:
 - SearchViewController: Controller responsible for handling the Search View functionality, that is: entering text, handling buttons, including a switch that allows switching between languages, and modifying request parameters.
 - PickViewController: Controller responsible for handling the Pick View functionality, that is: creating labels, handling the UIPickerView (scrolling selection list) and buttons, modifying request parameters.
 - NewsListViewController: Controller responsible for handling the functionality of the News List View, that is: building a request and sending it, then creating a NewsModel after receiving a correct return message or displaying a message after receiving an error.
- Server: One of two request is sent to the server (<https://newscatcher.p.rapidapi.com/v1/>) depending on which view was used by the user:
 - If the user used Search View, the request "search" is sent with the parameter "q" in which the entered phrase is located.
 - If the user used Pick View, the request "latest_headlines" is sent with the "topic" parameter in which the selected category is located.

Application layers and communication between them

- There are 4 layers in the application:

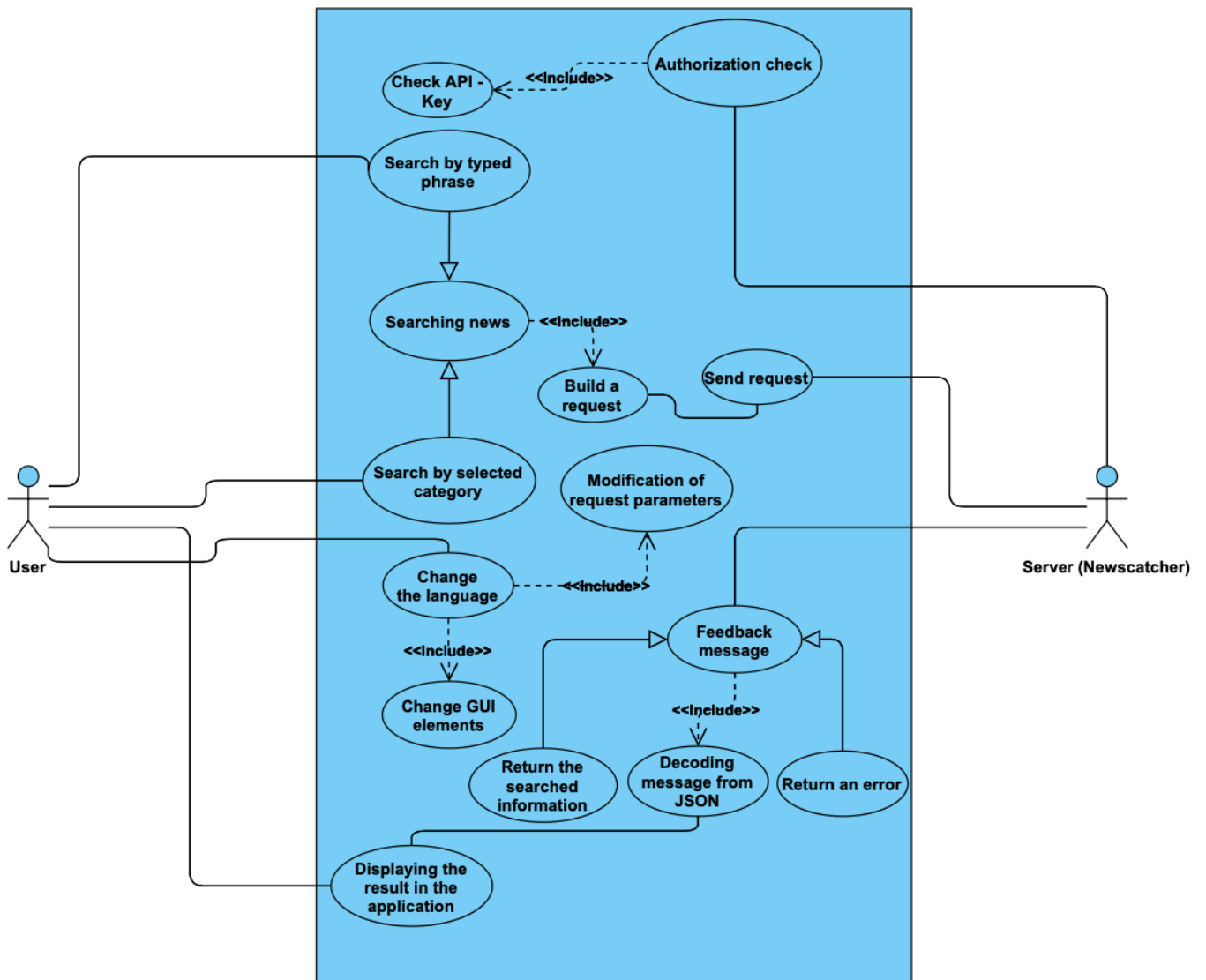


- The diagram of communication between the layers is shown below:



Use case and activity diagrams

- Use case diagram:



- Activity diagram:

