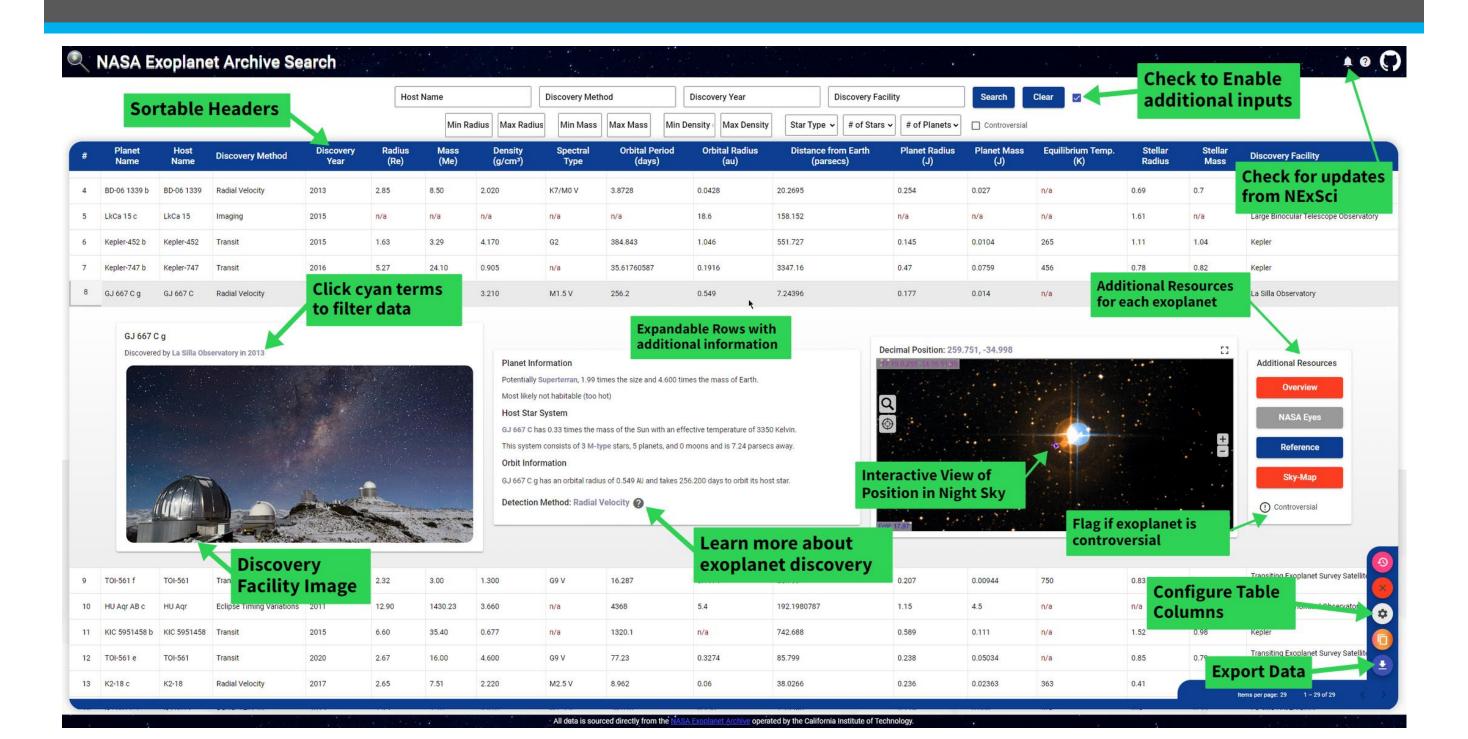# NASA Exoplanet Archive Search

By Adam Jarvis & William Leithauser | Advisor: Dr. Fred Annexstein | University of Cincinnati

Abstract

Our Angular-powered Exoplanet Archive Search simplifies the process of querying and retrieving data from NASA's exoplanet database, providing an intuitive and responsive experience with enhanced readability and user interaction.

User Interface

Data Source & Stack

- This web app was built using Angular 14 and is a Progressive Web App (PWA) that can be installed on most devices with a modern browser.
- This project utilizes Table Access Protocol (TAP) to request and return data from the Exoplanet Archive. TAP is a protocol developed by IVOA that allows access and querying of table data on remote servers, including astronomical data.
- This project also uses my Express.js CORS Proxy to add necessary headers to the HTTP response from the Exoplanet Archive

Angular Concepts & Features

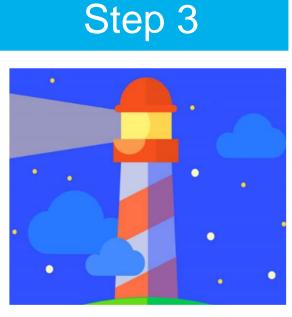| Components	Services	Directives
• Building blocks of Angular applications • Contains HTML templates, CSS styles, and JavaScript logic • Examples include Table component & InputBar component	• Provide functionality that can be shared across components • Used for tasks such as data retrieval • Examples include Data service and Loading service	• Create reusable behavior • Can apply behavior(s) to multiple existing elements or components • Example includes Sort directive in table

Development Process

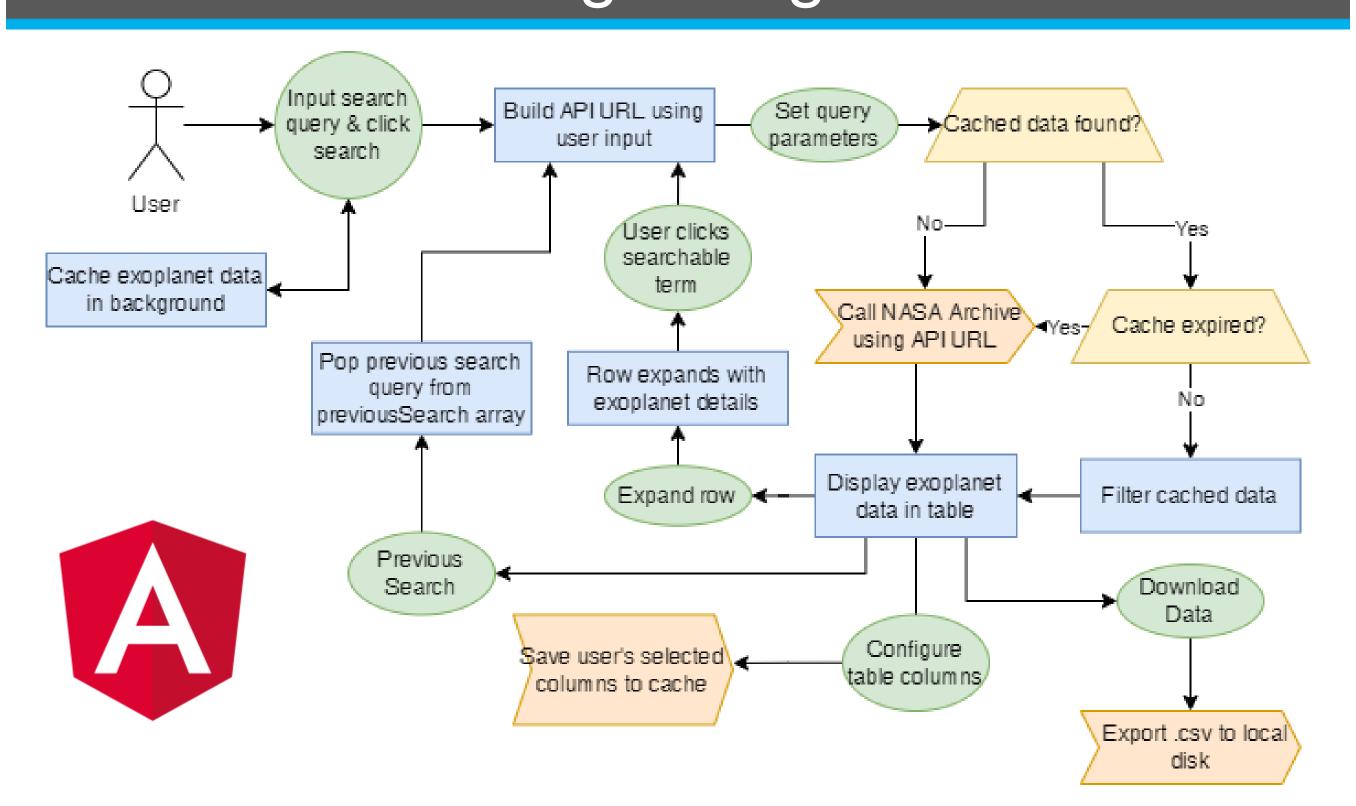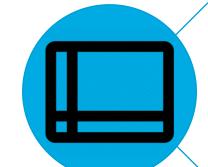| Step 1	Step 2	Step 3	Step 4
Initial web design and API testing	Build core UI elements such as input and table	Test and enhance performance and functionality	Implement responsive styling and dynamic sizing

Design Diagram

Core Project Layout

Name	Type	Function
App	Component	Root component and container
InputBar	Component	HTML, CSS, and JavaScript for input-bar elements
Data Service	Service	Used to call the Exoplanet Archive and return/cache data
Table	Component	Angular component for data table (includes table buttons & paginator)
Exodetail	Component	Angular component for expanded exoplanet row (includes Aladin visualization)
Loading & Interceptor	Service	HTTP interceptor used to conditionally render the loading spinner in AppComponent
Sort & Resize	Directive	Used to enable sorting and resizing on table headers in table component
Download	Service	Converts current table data to .csv format and exports to user's local disk

Implemented Features

Material Design
- Material Design is used throughout the web application giving it a consistent and modern look and feel

Interactive Table
- Table headers are sortable and resizable, with the ability to configure which columns are displayed.

Aladin Lite Sky Atlas
- Expand an exoplanet's row to view its position in Aladin Lite's interactive sky visualization.
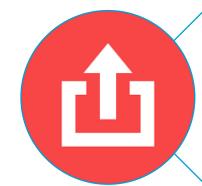
Progressive Web App Support
- PWA's are web applications designed to function like a native app on mobile devices and desktops.

Streamlined Data Retrieval
- With pagination and conditional rendering, the table loads instantly once exoplanet data is cached in the background.

Exportable Data
- Use the download button to export the table data to the local disk in .csv format

Conclusion & Final Thoughts

Overall, our Angular-powered Exoplanet Archive Search provides a user-friendly and efficient way to access and search NASA's exoplanet database and view the data in an interactive table.

With an intuitive interface, responsive design, and streamlined data retrieval, users can easily query and retrieve the data they need. We hope this web app will serve as a valuable resource for novice and expert astronomers alike.

Developers

Adam Jarvis
Computer Science

William Leithauser
Computer Science