Analyse and Design
of
Astronomy and Astrophysics Library

QSSL

Qazvin Islamic Azad University

Teacher : Dr. Seyed Hossein Mir Shojaei
Student : Ramtin Kosari

Fall 2024 (1403)


Somewhere,
Something
Incredible is
Waiting to be
Known.
- Carl Sagan

Table of Contents

An Introduction to QSS Library

QSSL is a comprehensive open source library project that serves as an all-in-one toolset for astrophysicists and concerned developers. It covers a wide range of astronomical subjects including calculations of orbits, celestial trigonometry, celestial mechanics, astrophysics and cosmology, also powerful tools for astronomical image processing based on artificial intelligence.

QSSL is designed to be a comprehensive toolset for astronomers, astrophysicists, developers, and space enthusiasts. The project's goal is to foster a community-driven environment that caters to a diverse range of users, from amateur astronomers to space industry professionals whether you're an amateur astronomer, a researcher, a gaming group who develops a Sci-fi astronomical game or also a space company that wants to calculate the celestial mechanics of orbits to find the cheapest way to reach mars !

Imagination is More Important than Knowledge ***- Albert Einstein***

A Brief Introduction to Potential Subsystems

QSSL will be organized into several key subsystems, each focused on different aspects of astronomy and astrophysics to ensure the library is comprehensive and flexible. These subsystems will enable users to perform a variety of tasks, from data analysis to complex simulations. Below are some of the potential modules :

- **Celestial Objects Module**
  Enables users to create hierarchical structures of the universe, including galaxies, star systems, planets, and other celestial bodies. This module provides a framework for representing celestial objects as interactive, interconnected entities for exploration and analysis.
- **Spherical Trigonometry Module**
  Provides tools for calculating angular distances, positions, and orientations of celestial objects on the celestial sphere. It supports essential computations for astrometry, including coordinate transformations, timings and spherical distance calculations.
- **Celestial Mechanics Module**
  Focuses on modeling the motions and interactions of celestial bodies under gravitational forces. It provides tools for calculating orbital dynamics, n-body problems, and predicting the movements of planets, moons, and other objects in space.
- **Astrophysics Module**
  Provides tools for analyzing astrophysical phenomena, such as stellar evolution, black holes, and cosmic radiation. It enables users to model complex physical processes and explore the underlying theories of the universe’s structure and behavior.
- **Cosmology Module**
  Offers tools for modeling and analyzing the large-scale structure of the universe, including the Big Bang, dark matter, and cosmic expansion. It enables users to simulate cosmological events and explore the evolution and dynamics of the universe on a grand scale.
- **Simulation Module**
  Enables the visual representation of astronomical simulations, turning complex data into interactive 3D models. It allows users to explore celestial events, such as star formations and orbital dynamics, through immersive visualizations including planetarium.

- **Data Analyzing Module**
  Provides tools for processing, analyzing, and interpreting large sets of astronomical data, such as photometric and spectroscopic data. Also it leverages machine learning algorithms to process and analyze data, enabling automated pattern recognition and anomaly detection. It enhances data interpretation by using AI to uncover hidden insights and optimize analysis workflows.
- **Image Processing Module**
  Provides advanced tools for analyzing and enhancing astronomical images, including noise reduction, feature extraction, and image calibration. It supports the manipulation of various astronomical image formats to enable detailed visualization and analysis of celestial objects.
- **Spectroscopy and Atomic Physics Module**
  Offers tools for analyzing the spectra of celestial objects, enabling the study of their composition, temperature, and motion. It also includes models for atomic transitions and line identification, essential for understanding stellar and planetary atmospheres.
- **Electromagnetic Theory and Quantum Physics Module**
  Provides tools for simulating and analyzing electromagnetic waves, quantum states, and particle interactions. It enables users to explore phenomena like wave-particle duality, quantum entanglement, and the behavior of light in various astrophysical contexts.
- **Thermodynamics Module**
  Offers tools for modeling and analyzing the thermal properties of celestial objects, including heat transfer, radiation, and energy balance. It supports the study of processes such as stellar cooling, black hole thermodynamics, and the behavior of matter under extreme conditions.
- **Nuclear Physics Module**
  Provides tools for simulating nuclear reactions, fusion, and fission processes in stellar environments. It enables the study of nucleosynthesis, radioactive decay, and the energy production mechanisms that power stars and other celestial bodies.
- **IoT and Embedded Systems Control Module**
  Provides tools for designing, controlling, and testing embedded systems and IoT devices. It includes communication protocols and control functions to manage electric circuits, actuators, and sensors, enabling seamless interaction with systems in real-time like rockets.

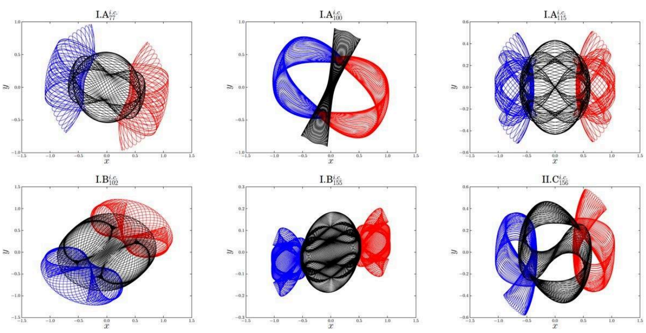
Presentation of Challenges and Limitations of the QSSL

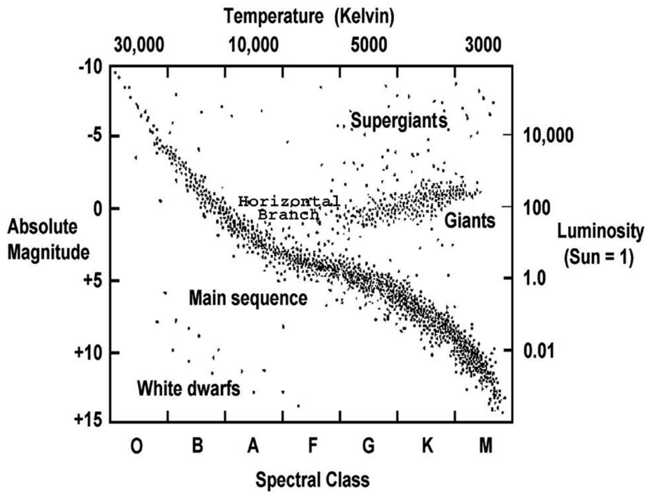
While the QSSL aims to provide comprehensive tools for researchers, educators, and enthusiasts, there are several challenges and limitations that must be addressed to ensure its effectiveness and widespread adoption :

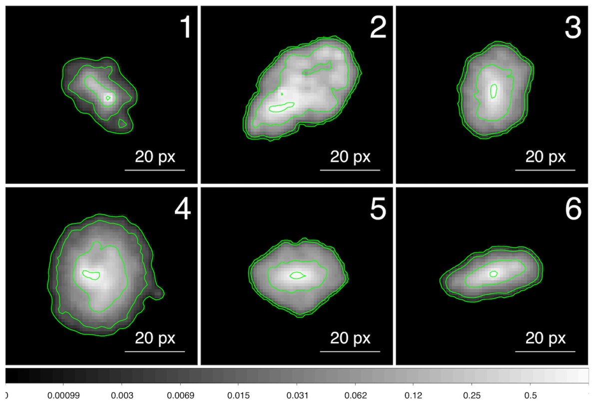
- **Scalability and Performance**
  As the library grows to handle increasingly large datasets and complex simulations, performance may be a concern. The system might struggle with processing large-scale astronomical data, running high-resolution simulations, or managing the computational demands of real-time data analysis.
- **Integration with Diverse Data Sources**
  The library is designed to work with a wide variety of astronomical data formats, but integrating and harmonizing data from different sources, instruments, and research methodologies can be challenging. Ensuring seamless compatibility with existing observatories, databases, and external tools is crucial for broader adoption.
- **User Accessibility and Expertise**
  While the library is intended to be user-friendly, some advanced features may require specialized knowledge in astrophysics or programming. The need for better documentation, user guides, and tutorials is vital to ensure that the system remains accessible to a wide range of users, from amateur astronomers to professional researchers.
- **Community Contributions and Collaboration**
  As an open-source, community-driven project, the quality and consistency of contributions can vary. Managing community contributions, ensuring code quality, and maintaining an organized development process are key challenges in sustaining long-term progress and ensuring a cohesive user experience.
- **Real-time Data Processing and Visualization**
  Many astronomical phenomena require real-time or near-real-time data processing. Developing efficient algorithms for real-time data acquisition, processing, and visualization remains a complex challenge, particularly when handling large amounts of dynamic observational data.

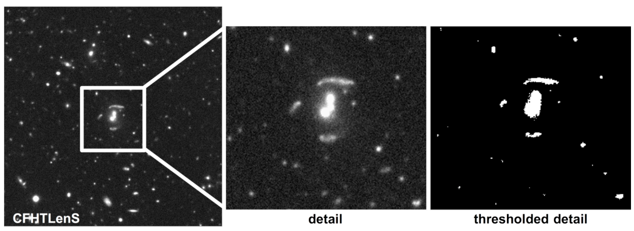
- **Hardware and Platform Compatibility**
  The library may need to interact with a range of hardware devices, such as telescopes, sensors, or even embedded systems. Ensuring broad compatibility with different operating systems, devices, and hardware interfaces presents technical challenges, particularly in maintaining cross-platform functionality.
- **Documentation and Learning Curve**
  Given the complexity of astronomical computations and simulations, some users may face a steep learning curve. Continuous effort is needed to improve the documentation, provide detailed examples, and create educational resources to make the library more accessible to users with varying levels of expertise.

By addressing these challenges, the project aims to evolve into a robust, reliable, and widely used resource in the field of astronomy and astrophysics. Ongoing community engagement, optimization, and user-focused development will be key to overcoming these limitations and ensuring the project's success.







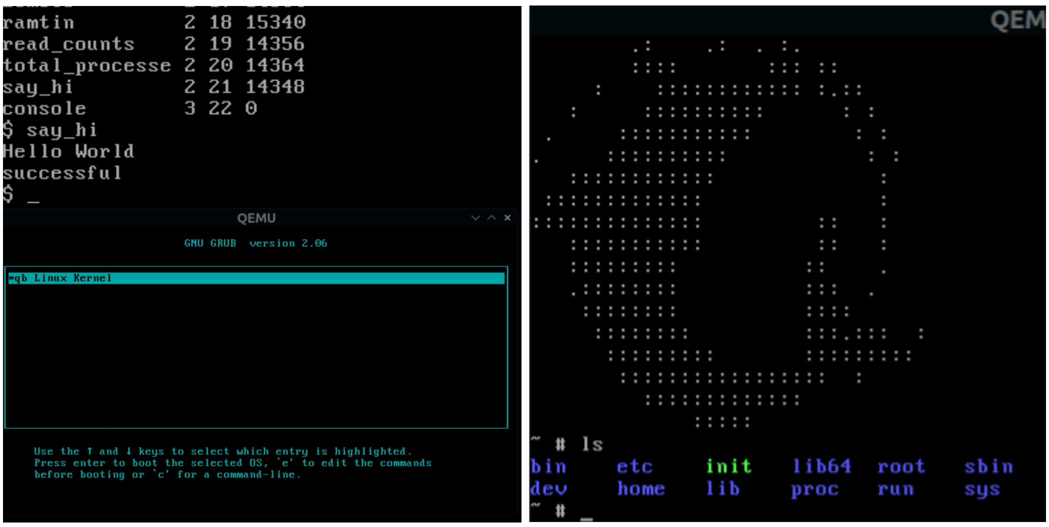


Presentation of Expectations and Features of the QSSL

As the open-source, community-driven astronomy and astrophysics library evolves, several expectations and features are envisioned to expand its capabilities, improve user experience, and further integrate into the global scientific community. These include :

- **Enhanced Scalability and Performance**
  The future system will be optimized to handle even larger datasets and more complex simulations, supporting high-resolution astronomical data and real-time processing without compromising performance. This will enable researchers to work with vast amounts of observational data and advanced models with ease.
- **Expanded Integration with External Data Sources**
  The library will evolve to seamlessly integrate with a broader range of astronomical databases, telescopes, and research platforms. Automatic data ingestion, synchronization with observatories, and support for new formats will make it easier to access, process, and analyze diverse astronomical data.
- **Improved User Accessibility and Educational Tools**
  The future system will include more intuitive user interfaces, enhanced documentation, and comprehensive tutorials for both novice users and experts. Interactive guides, educational resources, and a simplified installation process will help broaden the user base, making the library accessible to a global community of astronomers, students, and educators.
- **Real-time Data Analysis and Visualization**
  The system will include tools for real-time data analysis, allowing users to process live observational data, visualize celestial events as they occur, and simulate dynamic phenomena in real-time. This will be especially beneficial for users involved in time-sensitive astronomical research.
- **AI and Machine Learning Integration**
  Future iterations of the library will incorporate machine learning and AI-based tools to automate pattern recognition, anomaly detection, and predictive modeling in astronomical data. This will enable the system to identify trends, classify celestial objects, and optimize simulations with minimal human intervention.

- **Expanded Community Contributions and Collaboration**
  The open-source nature of the project will foster an even larger and more diverse community of contributors. The future system will feature a more organized contribution workflow, ensuring high-quality code, collaborative development, and active discussions that drive innovation and rapid improvements.
- **Cross-platform and Hardware Compatibility**
  The library will be optimized to work across various platforms (Windows, Linux, macOS) and be compatible with diverse hardware, from personal computers to embedded systems used in observational setups. Users will be able to integrate the system with telescope systems, sensors, and other astronomy-related devices with ease.
- **Advanced Simulation and Modeling Capabilities**
  The system will provide advanced modules for simulating astrophysical processes like star formation, galaxy dynamics, and black hole interactions. It will allow users to run highly detailed, large-scale simulations and explore new phenomena in the universe, all with higher accuracy and faster processing times.
- **Sustainability and Long-term Support**
  The project will focus on long-term sustainability by ensuring ongoing development and maintenance. With robust community-driven support, regular updates, and feature expansions, the system will remain a reliable tool for the global astronomy and astrophysics community.



Introduction to Skills and Expertise

The success of the QSSL open-source astronomy and astrophysics project relies on a diverse set of skills and expertise contributed by the global community. The following areas of expertise are essential for the development, enhancement, and long-term sustainability of the system :

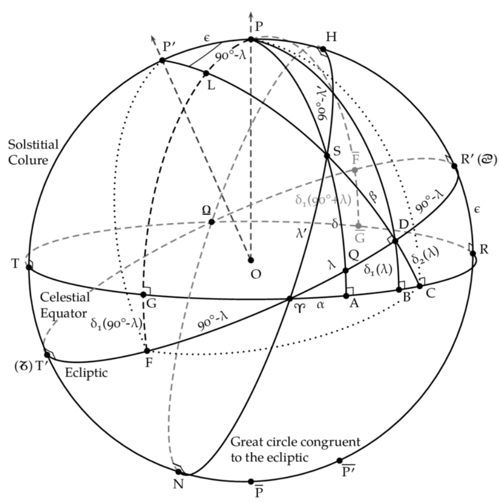
- **Astronomy and Astrophysics Knowledge**
  A deep understanding of astronomical phenomena, celestial mechanics, cosmology, and astrophysics is fundamental. This expertise is necessary to ensure that the system accurately models and simulates celestial bodies, cosmic events, and related physical processes.
- **Software Development and Programming**
  Proficiency in programming languages such as C++ is essential for the development and maintenance of the library. Developers with experience in object-oriented programming, algorithm design, and optimization are key to building this fast, efficient, and scalable system.
- **Data Analysis and Machine Learning**
  Expertise in data science, statistical analysis, and machine learning is crucial for developing tools that can handle large-scale astronomical datasets, automate pattern recognition, and make predictions based on observational data.
- **Numerical Simulation and Modeling**
  Specialists in numerical methods, computational physics, and simulation algorithms will contribute to the development of accurate and efficient models for simulating astrophysical phenomena, such as stellar dynamics, galaxy formation, and black hole interactions.
- **User Interface and User Experience (UI/UX) Design**
  Skilled UI/UX or graphical designers are needed to make the system intuitive and user-friendly. Their expertise in designing clear, functional, and visually engaging interfaces will ensure that the library is accessible to users with varying levels of experience, from beginners to professionals.
- **Astronomical Instrumentation and Hardware Integration**
  Knowledge of how to interface the library with physical astronomical instruments, such as telescopes, sensors, and embedded systems, is essential for expanding the system's functionality and enabling real-time data collection and analysis.

- **Community Management and Open-Source Collaboration**
  Effective communication, project management, and collaboration skills are essential for fostering a thriving open-source community. Expertise in managing contributions, conducting code reviews, and organizing documentation will ensure smooth project operations and promote active involvement
- **Documentation and Educational Content Creation**
  Experts in writing clear and concise documentation, tutorials, and educational resources are vital for ensuring that the system is accessible to a broad audience. This includes creating guides for new users, developing teaching materials, and ensuring up-to-date documentation as the project evolves.

By combining these diverse areas of expertise, the project will continue to evolve, providing high-quality tools for researchers, educators, and astronomy enthusiasts worldwide. Collaboration among individuals with varied skill sets will be key to the continued growth and success of this community-driven initiative.

$$\left(H^2 - \frac{8}{3}\pi G\rho\right)R^2 = -kc^2$$

$$\left[\left(\frac{1}{R}\frac{dR}{dt}\right)^2 - \frac{8}{3}\pi G\rho\right]R^2 = -kc^2$$



Introduction to Methods and Tools

The following methods and tools are central to the development, maintenance, and community engagement of the QSSL. These tools facilitate design, code development, testing, simulation, and data analysis :

- **C++ :** Used for high-performance components and complex simulations and best programming language for developing libraries.
- **Doxygen :** Automatically generates documentation from the source code, ensuring clear code descriptions.
- **Git and Github :** Provides version control and collaboration for code development and project management.
- **PostgreSQL :** A relational database for storing and querying structured astronomical data such as celestial objects catalogs.
- **RUP :** A good structured, iterative development methodology that guides project phases.
- **Discord :** A platform for real-time communication and collaboration within the community.
- **OpenGL :** Used for rendering 2D and 3D visualizations of celestial objects and simulations.
- **OpenCV :** A computer vision library for processing and analyzing celestial images and applying filters.
- **Redis :** An in-memory database for fast data caching and improving system performance.
- **Qt :** Framework for developing cross-platform graphical user interfaces (GUIs) used for tools applications and GUI related programs.
- **Python Libraries (Numpy, Scipy, Matplotlib) :** Python, with its powerful libraries like NumPy, SciPy, and Matplotlib, forms the backbone of the computational and AI aspects of the project. These libraries are essential Machine Learning Algorithms
- **UML :** Used for creating diagrams to model and document the system's architecture.
- **draw.io :** A structured, iterative development methodology that guides project phases.

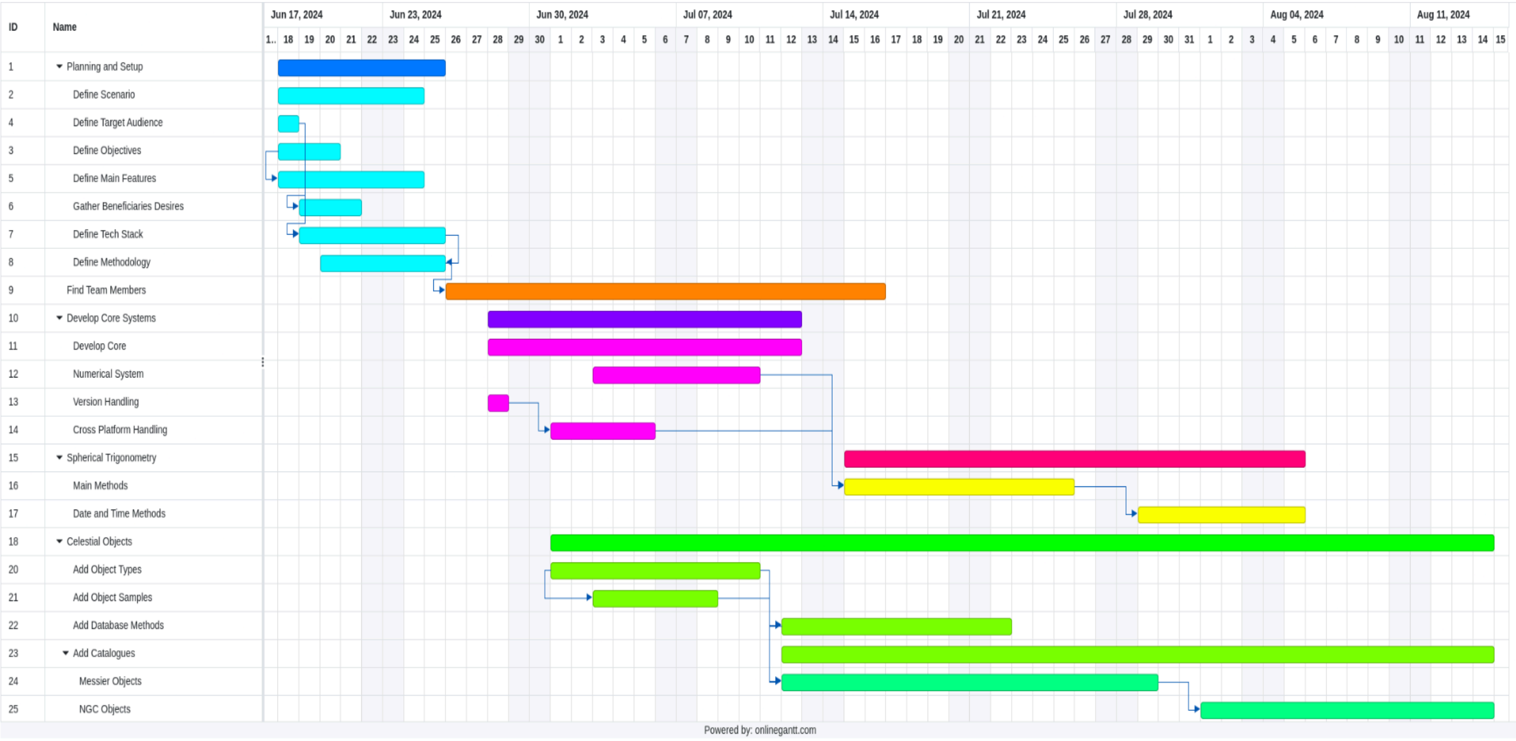
Cost Estimation

The following table shows cost estimation of QSSL :

Cost Item	Description	Cost Estimation	Remarks
Main Development Team	Time for main developers to create base and main library structures and sub modules	150 $ - 300 $ according to skills and tasks assuming for 3 main developers monthly	Depends on developer's experience and project complexity
Open-Source Development	Time for open-source contributors to work on smaller tasks, projects and patches	50 $ - 500 $ awards or salary of open-source developers any time	Depends on awards, competitions and open-source sub project contributors
Website	Cost of Front-end, Back-end and Domain reservation for library's online web page	30 $ - 300$ cost of domain, development and maintenance	based on Front-end and Back-end development, SSL certificates, maintenance, domain and hosts costs
Subscriptions	cost of necessary subscriptions for team managements like Github Enterprise or Discord Premium	40 $ Discord Premium and Github Enterprise Costs Monthly	depending on the specific tools or services needed like Discord or Github products
Platform Licenses	cost of Tools and Platforms necessary license	Free	maybe free because mentioned tools are free and open-source
Testers Team	Time for testers to test functions and usage of the library in different conditions and environments	20 $ - 50 $ each time conducting test tasks	depends on the complexity of testing (manual vs. automated) and the number of testing phases required or number of required testers
Online Database	Cost of online server where the library is be able to access databases of celestial objects via internet	40 $ - 100 $ AWS Linux Server Monthly	based on database size, query complexity, and storage requirements for large datasets.
Miscellaneous Costs	Unforeseen costs like additional software, or tools that may be required for the library	0 $ - 200 $	Buffer for unexpected expenses during the course of the library
Total	**-**	**~200 $ - 400 $**	**-**

Time and Scheduling Estimation

The following Gantt Diagram shows time and scheduling estimation of QSSL Nova Version (First Release Version of the QSSL) :



File of this Gantt Chart exists in projects assets directory or attached in email, in this Gantt Chart for Nova version of QSSL it takes about 2 months to be done.

Beneficiaries Desires

Several questions about features and services have been asked from users, including astronomers, professors of astrophysics and cosmology and …

Here are their answers :

1. **Ramtin Kosari** - Computer Engineer, Amateur Astrophysicist
   "I need a reliable tool to predict celestial events accurately for my research on planetary transits. The current tools I use are either outdated or too complicated."
2. **Ramtin Kosari** - Computer Engineer, Amateur Astrophysicist
   "As a student preparing for the Astrophysics Olympiad, I need access to a comprehensive library of solved problems and detailed solutions. Tools that simulate real Olympiad scenarios and offer step-by-step problem-solving strategies would be incredibly helpful for my preparation."
3. **Ehsan Ebrahimian** - Sharif PhD Student in Cosmology, Silver Medal in IOAA
   "In my research on celestial mechanics, I need robust tools to solve the equations governing the stable orbits at Lagrange points L4 and L5. A library that provides precise computational methods and visualizations of these stable orbits would be crucial for advancing my studies."
4. **Ehsan Ebrahimian** - Sharif PhD Student in Cosmology, Silver Medal in IOAA
   "I work on developing algorithms for improving astronomical images. Access to a library with advanced noise reduction techniques such as variance stabilizers, wavelet noise filters, and Anscombe transformations would greatly aid in refining the quality of images captured by low-sensitivity cameras."
5. **Lida Molla Mohyeddin** - Amateur Astronomer
   "As an amateur astrophotographer using a weak camera, I need powerful image processing algorithms to enhance the quality of my star images. Tools that include variance stabilizers, wavelet noise filters, and Anscombe transformations to remove noise would help make my astrophotography much clearer and more detailed."
6. **Ehsan Rostami Darestani** - Aerospace Engineer, Game Developer, Musician
   "I am developer of a space-themed game that requires accurate calculations of celestial mechanics to simulate realistic space environments and spacecraft navigation. A library that provides

detailed algorithms for orbital dynamics, gravitational interactions, and real-time simulation of celestial movements would be essential for creating an engaging and scientifically accurate game experience.”

7. **Ata Moradi** - Sharif Master Student in Mechanics, Silver Medal in IOAA

   “In my work with satellite data, I need advanced tools for handling and visualizing satellite orbits. Features that allow me to accurately model, simulate, and visualize the trajectories of satellites in 3D would greatly enhance my ability to analyze orbital mechanics and predict satellite positions”

8. **Cosmologist**

   “I need creative tools for visualizing and analyzing large-scale structures in the universe. Features that allow customization of visualization parameters and interactive exploration of data would enhance my research presentations.”

9. **Chemistry Teacher**

   “In my chemical physics research, I often use spherical trigonometry to calculate angles between atoms in molecules like CH4. Having a library with precise trigonometric functions tailored for such calculations would be a game-changer.”