

SPACE VIEWER

Earth, Moon, Mars and beyond, through NASA's lens.



Earth from deep space · NASA photograph. The web version of this page includes an interactive NASA image viewer and a link to NASA's official ISS stream.

NASA COMMUNITY COLLEGE AEROSPACE SCHOLARS · STUDENT CHALLENGE FY2026-27

Exploring the Software Behind Space Missions

Ground Systems Software Engineering

Khachatur Hakobyan · Student, Glendale Community College

Independent student portfolio for the NASA Community College Aerospace Scholars (NCAS) Student Challenge. Not an official NASA website.

Childhood

University

Harvard CS50

Meta iOS

Professional Experience

GCC

NASA NCAS

The Future

Mission-support software, built on a production foundation.

Ground Systems Software Engineering is the career focus of this portfolio: the software on Earth that receives spacecraft telemetry, prepares and sends commands, and gives mission teams a clear view of their spacecraft.^{1, 4}

The foundation is a Bachelor's degree in Computer Science and Applied Mathematics from Yerevan State University and professional iOS engineering since 2014, most recently as a Senior iOS Engineer, with iOS team leadership at four companies. That work centered on software architecture, shared SDKs, Swift and C++ debugging, automated testing and CI/CD, the disciplines that keep complex software dependable.

Current study at Glendale Community College strengthens the systems side. Programming in C/C++ and Introduction to Programming in Java are complete; Discrete Structures and Computer Architecture and Assembly Language are in progress. Harvard CS50 and the Meta iOS Developer Professional Certificate add structured, continuous learning.

Through the NCAS Student Challenge, this portfolio maps that coursework and experience to NASA's mission workforce: how ground systems support operations, which skills the work requires, and the education still ahead, with graduate study in Systems Engineering as a long-term goal.

PATHWAY The target role and the foundation behind it, stated once and carried through every section.

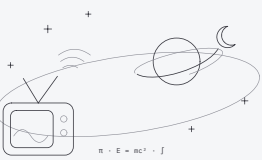
CAREER JOURNEY · 01-07

From Curiosity to Space Exploration

Seven chapters, from early wonder to a focused aerospace pathway.

01

FIG. 01



CHAPTER 01 · CHILDHOOD

Wonder Beyond Earth

A childhood fascination with space, planets, stars and the universe.

Curiosity reached the mathematics and physics that explain them, and the inner workings of computers, televisions, radios and electronics.

Space Planets Stars Universe Mathematics Physics Computers Electronics

02

FIG. 02



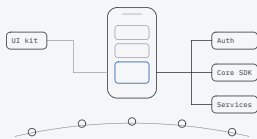
CHAPTER 02 · ACADEMIC FOUNDATIONS

Knowledge Without Boundaries

Formal computer science education, followed by continuous learning.

Yerevan State University	Bachelor's degree
Computer Science & Applied Mathematics · studies Sep 2010-Sep 2017	
Harvard CS50	Course
Introduction to Computer Science	
Meta	Certificate
iOS Developer Professional Certificate	

FIG. 03



CHAPTER 03 · PROFESSIONAL EXPERIENCE · SINCE 2014

Designing Software at Scale

iOS engineering since January 2014, then senior engineering and iOS team leadership across five companies, focused on architecture, shared platforms and reliable delivery.

KSF Technologies AG

Senior iOS Engineer

Nov 2021 – Present

Crypmoji

iOS Team Lead (Senior iOS Engineer)

Feb 2021 – Nov 2021

WiredMates

iOS Team Lead (Senior iOS Engineer)

Mar 2020 – Feb 2021

Xamlab

iOS Team Lead (Senior iOS Engineer)

Jan 2019 – Mar 2020

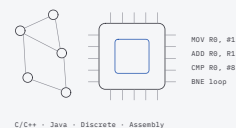
Arm-BoldMind

iOS Engineer → iOS Team Lead (Promoted)

Jan 2017 – Jan 2019

[Three roles in detail below →](#)

FIG. 04



CHAPTER 04 · GLENDALE COMMUNITY COLLEGE

Expanding the Foundation

U.S. community college study focused on systems foundations.

Glendale, California, United States · Computer Science / STEM Coursework · Ongoing studies

Pursuing computer science and STEM coursework at Glendale Community College, building on professional software engineering experience through studies in Java, C/C++, discrete structures, computer architecture, and Assembly language. Integrating computing and Earth science concepts to explore connections between academic coursework, reliable software systems, computational problem-solving, and NASA-related aerospace career pathways through the NASA Community College Aerospace Scholars (NCAS) Student Challenge.

CS/IS 135 · Programming in C/C++

COMPLETED

CS/IS 112 · Introduction to Programming Using Java

COMPLETED

CS/IS 125 · Discrete Structures for Computing

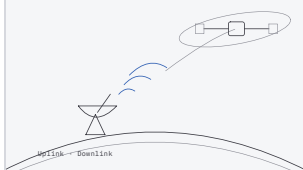
IN PROGRESS

CS/IS 165 · Computer Architecture and Assembly Language

IN PROGRESS

[How each course maps to NASA work →](#)

FIG. 05

CHAPTER 05 · NASA NCAS · **NOW**

A Path Toward Space

Participant in the NASA Community College Aerospace Scholars Student Challenge, FY2026–27.

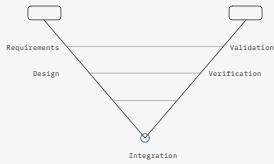
NASA Community College Aerospace Scholars (NCAS) · Student Challenge · 2026–2027 academic cycle

Career research centered on Ground Systems Software Engineering: the software that connects spacecraft with the teams who operate them.

[NASA overview](#)
[NASA careers](#)
[O*NET research](#)
[Career portfolio](#)

06

FIG. 06



CHAPTER 06 · FUTURE CAREER · GOAL

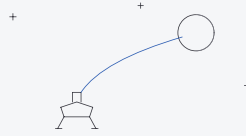
Architecting Complex Systems

Long-term direction: systems engineering and architecture.

Requirements, integration, verification and reliability across complex systems, with graduate study in Systems Engineering as a goal.

07

FIG. 07



CHAPTER 07 · THE VISION · VISION

Beyond the Horizon

Software for intelligent mission control, autonomous spacecraft and resilient space systems.

The long-range aim behind every chapter above.

PATHWAY Each chapter adds a layer: curiosity, computer science, production engineering, systems coursework and NASA career research.

ACADEMIC & SKILL ALIGNMENT

From Coursework to Mission Systems

GCC coursework builds technical skills; each skill maps to a ground-systems task. Select any course, skill or task to trace its path.

GCC COURSE



TECHNICAL SKILL



GROUND-SYSTEMS TASK

COMPLETED

CS/IS 135 · Programming in C/C++

C/C++ systems programming

Coursework + professional

- Telemetry processing & ground data systems
- Command preparation & uplink

Debugging & memory safety

Coursework + professional

- Anomaly investigation & reliability

COMPLETED

CS/IS 112 · Introduction to Programming (Java)

Object-oriented design

Coursework + professional

- Mission control displays & tools
- Command preparation & uplink

Software architecture & shared services

Professional

- Telemetry processing & ground data systems
- Mission control displays & tools

IN PROGRESS

CS/IS 125 · Discrete Structures

Discrete math & algorithms

In progress

- Command preparation & uplink
- Verification & validation

Testing, CI/CD & verification

Professional

- Verification & validation

IN PROGRESS

CS/IS 165 · Computer Architecture and Assembly

Computer architecture & data representation

In progress

- Telemetry processing & ground data systems
- Anomaly investigation & reliability

Debugging & memory safety

Coursework + professional

- Anomaly investigation & reliability

BEYOND GCC COURSEWORK

Professional experience · Harvard CS50

Technical leadership & code review

Professional

- Verification & validation
- Mission control displays & tools

Python scripting & data tools

Coursework

- Telemetry processing & ground data systems
- Mission control displays & tools

Technical Skills

EVIDENCE AND NASA RELEVANCE FOR EACH

C/C++ systems programming

PROFESSIONAL

COURSEWORK

EVIDENCE

CS/IS 135 (completed); Swift-C++ interoperability work at Crypmoji.

NASA USE

Telemetry and command software; C++ also underpins NASA's F' flight software framework.⁵

Object-oriented design

PROFESSIONAL

COURSEWORK

EVIDENCE

CS/IS 112 Java (completed); protocol-oriented Swift design at KSF and Xamlab.

NASA USE

Modular, maintainable ground applications and operations tools.

Software architecture & shared services

PROFESSIONAL

EVIDENCE

Core SDK and authentication redesign at KSF; modular MVVM at Xamlab.

NASA USE

Multimission ground software, such as NASA's AMMOS product line.³

Debugging & memory safety

PROFESSIONAL

COURSEWORK

EVIDENCE

Swift-C++ ownership defect resolved at Crypmoji; memory topics in CS50 and CS/IS 135.

NASA USE

Anomaly investigation and reliable long-running services.

Testing, CI/CD & verification

PROFESSIONAL

EVIDENCE

Unit testing and Azure DevOps CI/CD at Xamlab; review standards across teams.

NASA USE

Software assurance across the ground-systems lifecycle.⁹

Discrete math & algorithms

IN PROGRESS

EVIDENCE

CS/IS 125 (in progress); algorithms and applied mathematics at YSU.

NASA USE

Reasoning about command sequences, scheduling and correctness.

Computer architecture & data representation

IN PROGRESS

EVIDENCE

CS/IS 165 (in progress); Assembly coursework at YSU.

NASA USE

Decoding binary telemetry and understanding hardware-software interfaces.

Technical leadership & code review

PROFESSIONAL

EVIDENCE

iOS Team Lead at four companies; mentoring and code review standards.

NASA USE

Multi-team design work; conferring with engineers and analysts is a core task of the occupation.⁶

Python scripting & data tools

COURSEWORK

EVIDENCE

Harvard CS50 (Python, SQL and web programming units).

NASA USE

Ground tooling and data handling; F's ground data system layer, fprime-gds, is a Python package.¹¹

PATHWAY Every course and skill above points to a specific ground-systems task.

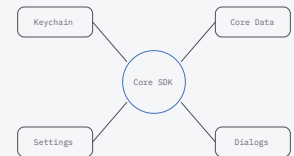
Experience That Transfers

Three engineering stories, each with its link to mission-support software. Every sentence below is part of the submitted portfolio.

01 · ARCHITECTURE

Authentication redesign and a shared Core SDK

KSF Technologies AG · Senior iOS Engineer · Zürich (remote) · Nov 2021 – Present



EXPERIENCE

Redesigned the login and authentication system of a large iOS application, replacing tightly coupled code with protocol-oriented, service-based modules. Built a reusable Core SDK that centralizes Keychain storage, Core Data, settings and dialog services in one shared layer. Applied SOLID, DRY and KISS principles to keep module boundaries clear and code reusable. Authentication-related defects decreased, and duplicated code fell as features moved onto the shared SDK.

NASA & INDUSTRY ALIGNMENT

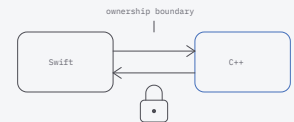
Ground systems are assembled from shared services; NASA's AMMOS, for example, provides multimission software for mission operations.³ Designing one SDK for many features follows the same principle: solve a problem once, test it well and reuse it safely. Careful credential handling and secure storage build habits that matter wherever commands and data must stay protected. Clear module boundaries also make software easier to verify, which reduces risk for the teams that depend on it.

Swift Protocol-oriented design SOLID Keychain Core Data

02 · SYSTEMS DEBUGGING

Swift-C++ memory ownership and a four-day prototype

Crypmoji · iOS Team Lead (Senior iOS Engineer) · Los Angeles · Feb 2021 – Nov 2021



EXPERIENCE

Led iOS work on a proof of concept for live encrypted messaging after comparing three delivery strategies: simulation, full integration and a phased MVP. Delivered a working demonstration within a fixed four-day window. Diagnosed a long-standing Swift-C++ memory ownership defect and removed unsafe handoffs between the two languages. Fixing the root cause avoided a full rewrite of the affected code.

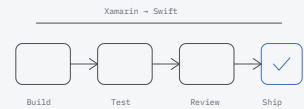
NASA & INDUSTRY ALIGNMENT

Mission software often spans several languages and layers; NASA's F-18 flight software framework, for example, is built in C++.⁵ Ownership errors at a language boundary are hard to detect and may appear only under load, so disciplined root-cause analysis matters. The same method of reproducing, isolating and fixing the cause applies to investigating anomalies in operations. Choosing a delivery strategy under a fixed four-day limit reflects the trade-offs that mission schedules demand.

Swift C++ Memory ownership Encryption Rapid prototyping

Xamarin to Swift migration with testing and CI/CD

Xamlab · iOS Team Lead (Senior iOS Engineer) · Los Angeles · Jan 2019 – Mar 2020



EXPERIENCE

Led the migration of a ticket-validation app from Xamarin and C# to native Swift with modular MVVM and protocol-oriented design. Introduced comprehensive unit testing and set up CI/CD in Azure DevOps with stronger pull-request standards. Ticket validation became faster and more reliable, and the app's App Store rating improved. Mentored engineers on testing and code review practices.

Swift

MVVM

Unit testing

Azure DevOps CI/CD

Mentoring

NASA & INDUSTRY ALIGNMENT

Ground software must behave correctly every time, so automated tests and gated releases are central to how it is built and changed. Migrating a working system without losing behavior resembles modernizing long-lived ground systems that missions still rely on. Fast, accurate validation at the point of use parallels processing and decoding telemetry correctly as it arrives.¹ Mentoring on review and testing spreads the verification culture that complex, multi-team missions need.

PATHWAY Each project ends with its direct link to mission-support software.

CAREER PATHWAY

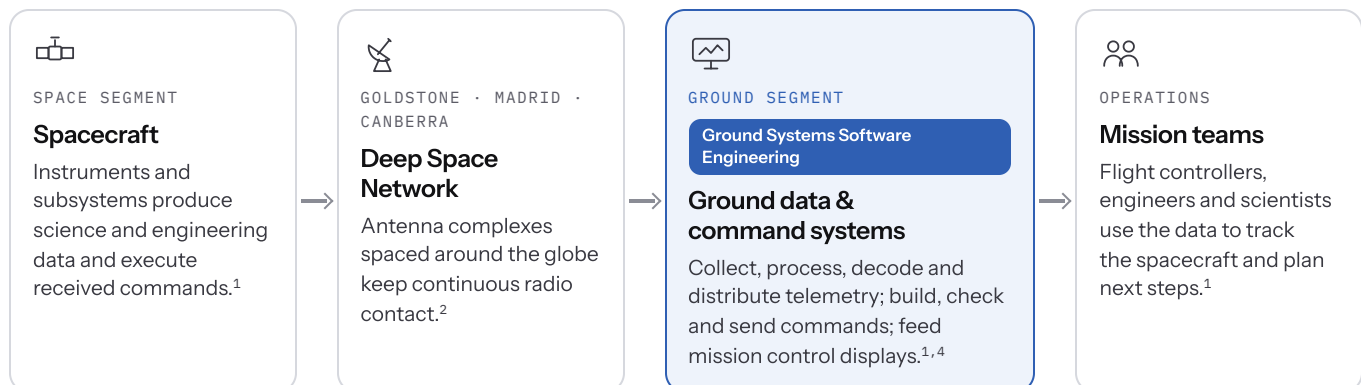
Ground Systems Software Engineering

The software on Earth that keeps missions connected, observable and safe.

Telemetry carries a spacecraft's science and engineering data to Earth, where ground systems collect, process, decode and distribute it. Commands travel the other way, as coded files the spacecraft executes.¹

NASA's Deep Space Network keeps continuous contact through antenna complexes near Goldstone, California; Madrid, Spain; and Canberra, Australia.² On the ground, multimission software such as NASA's AMMOS³ and mission control frameworks such as Open MCT, developed at NASA's Ames Research Center,⁴ turn that data into decisions.

Follow the signal



Telemetry path: Spacecraft → Deep Space Network → Ground systems → Mission teams.

Telemetry processing

Command preparation & uplink

Tracking & navigation data

Mission control displays

Multimission services

Systems foundations

Discrete Structures and Computer Architecture and Assembly Language at GCC; NCAS career research.

Deeper computer science and math

Data Structures, Calculus and Linear Algebra appear in the GCC academic plan; not yet enrolled.

Systems Engineering

Graduate study in Systems Engineering and ground systems software roles in aerospace.

Occupation research**O*NET OCCUPATION****Software Developers · 15-1252.00**

Research, design, and develop computer and network software or specialized utility programs.⁶

- Analyze user needs and software requirements to determine feasibility of design within time and cost constraints.
- Develop or direct software system testing or validation procedures, programming, or documentation.
- Modify existing software to correct errors, adapt it to new hardware, or upgrade interfaces and improve performance.

Job Zone Four: considerable preparation needed · Bright Outlook occupation⁶

NASA-EQUIVALENT ROLE**Computer Engineer, AST – Ground Data Systems**

Series 0854, Computer Engineering · NASA Goddard Space Flight Center, Greenbelt, Maryland.^{9, 10}

- Safety, mission and software assurance across the ground-systems lifecycle.⁹
- Security planning for a Near Earth Network system that supports satellite ground stations.¹⁰
- Qualifying degrees include computer science with specified mathematics, statistics and computer science coursework, including calculus.¹⁰

Both announcements are senior GS-15 positions: a long-term destination for this pathway.

REAL-TIME STATUS**DSN Now**

Which antennas are talking to which spacecraft, updated every few seconds.⁷

SIMULATION**Eyes on the Solar System**

NASA's 3D view of planets and spacecraft, built from mission data.⁷

CALCULATED DATA**JPL Horizons**

Computed positions and trajectories of solar system objects.⁸

OPEN-SOURCE SOFTWARE**Open MCT**

NASA's mission control framework for telemetry visualization.⁴

PATHWAY Coursework → skills → ground systems role → graduate study: one continuous line.

CHAPTER 07 · THE VISION

Beyond the Horizon

From childhood wonder to a future of discovery and innovation.

Intelligent mission control

Autonomous spacecraft

Resilient space systems

Lunar terrain · NASA

Prepared for What's Next

An aerospace-aligned resume, rendered on the page and printed as its own sheet in the PDF.

Khachatur Hakobyan

Senior iOS Engineer · Ground Systems Software Engineering pathway · NCAS Student Challenge FY2026–27

Los Angeles, California, USA · khachatur.hakobyan2023@gmail.com ·

linkedin.com/in/khachatur-hakobyan

PROFILE

Senior iOS engineer and technical leader: professional iOS engineering since 2014, including iOS team leadership at four companies. Focused on software architecture, reliability, cross-language debugging and testing; preparing for aerospace ground systems work through community college study.

EXPERIENCE

KSF Technologies AG Nov 2021 – Present

Senior iOS Engineer

Redesigned authentication with protocol-oriented, service-based modules; built a reusable Core SDK.

Crypmoji Feb 2021 – Nov 2021

iOS Team Lead (Senior iOS Engineer)

Delivered an encrypted-messaging prototype in four days; resolved a Swift-C++ memory ownership defect.

WiredMates Mar 2020 – Feb 2021

iOS Team Lead (Senior iOS Engineer)

Built a reusable UI system; integrated Stripe and Apple Pay checkout.

Xamlab Jan 2019 – Mar 2020

iOS Team Lead (Senior iOS Engineer)

Led the Xamarin-to-Swift migration with modular MVVM, unit testing and CI/CD.

Arm-BoldMind Jan 2017 – Jan 2019

iOS Engineer → iOS Team Lead (Promoted)

Delivered iOS apps with MVC/VIPER and Core Data; reduced crashes through analysis and refactoring.

EDUCATION

Glendale Community College Sep 2024 – Present

College

Completed: Introduction to Programming Using Java; Programming in C/C++. In progress: Discrete Structures for Computing; Computer Architecture and Assembly Language.

Yerevan State University Sep 2010 – Sep 2017

Bachelor's degree, Computer Science & Applied Mathematics.

Harvard CS50 Sep 2024 – Aug 2025

Introduction to Computer Science.

Meta Sep 2023 – Sep 2024

iOS Developer Professional Certificate.

TECHNICAL SKILLS

Swift, Objective-C, C++ (professional); C, Java, Python (coursework) · Software architecture: modular, MVVM, protocol-oriented · Debugging and memory safety · Unit testing, CI/CD, code review · Discrete structures, computer architecture (in progress)

PROGRAM

NASA Community College Aerospace Scholars Student Challenge, FY2026–27: career research in Ground Systems Software Engineering.

References

- 1 **NASA. "What Does the DSN Do?"** Space Communications and Navigation. Accessed Oct 19, 2026.
<https://www.nasa.gov/directorates/somd/space-communications-navigation-program/dsn-functions/>
- 2 **NASA. "Where is the DSN Located?"** Space Communications and Navigation. Accessed Oct 19, 2026.
<https://www.nasa.gov/directorates/somd/space-communications-navigation-program/dsn-complexes/>
- 3 **NASA AMMOS. "Frequently Asked Questions."** Advanced Multi-Mission Operations System. Accessed Oct 19, 2026.
<https://ammos.nasa.gov/faq>
- 4 **NASA Ames Research Center. "Open MCT."** GitHub repository README. Accessed Oct 19, 2026.
<https://github.com/nasa/openmct>
- 5 **NASA Jet Propulsion Laboratory. "F" (F Prime).** GitHub repository README. Accessed Oct 19, 2026.
<https://github.com/nasa/fprime>
- 6 **O*NET OnLine. "Summary Report for 15-1252.00 – Software Developers."** National Center for O*NET Development. Accessed Oct 19, 2026.
<https://www.onetonline.org/link/summary/15-1252.00>
- 7 **NASA Science. "Eyes."** DSN Now and Eyes on the Solar System. Accessed Oct 19, 2026.
<https://science.nasa.gov/eyes/>
- 8 **NASA JPL Solar System Dynamics. "Horizons System."** Accessed Oct 19, 2026.
<https://ssd.jpl.nasa.gov/horizons/>
- 9 **USAJOBS. "Computer Engineer, AST – Ground Data Systems."** NASA Goddard Space Flight Center, announcement 798441200 (closed Jul 15, 2024). Accessed Oct 19, 2026.
<https://www.usajobs.gov/job/798441200>
- 10 **USAJOBS. "Computer Engineer, AST, Ground Data Systems."** NASA Goddard Space Flight Center, announcement 805985200 (closed Aug 30, 2024). Accessed Oct 19, 2026.
<https://www.usajobs.gov/job/805985200>
- 11 **Python Package Index. "fprime-gds: F Prime Flight Software Ground Data System layer."** Accessed Oct 19, 2026.
<https://pypi.org/project/fprime-gds/>

Image credits. Earth from deep space, lunar terrain and Apollo astronaut bootprint: NASA, via NASA 3D Resources (github.com/nasa/NASA-3D-Resources). Mars: rendered for this page from NASA's global Mars color mosaic (same source). Hubble eXtreme Deep Field: NASA, ESA, G. Illingworth, D. Magee and P. Oesch (University of California, Santa Cruz), R. Bouwens (Leiden University) and the HUDF09 Team.

Independent student portfolio for the NASA Community College Aerospace Scholars (NCAS) Student Challenge. Not an official NASA website. No NASA insignia are used; NASA imagery is credited above.

khakobyan.com/ncas · Khachatur Hakobyan