Team name: SecureSpace
Team members: Michael Sammueller, Bradley Graham, Tomas Mestanza, Rachel Doherty

SSDCD_PCOM7E: Group Design Document

Wordcount: 1095

Background

The International Space Station (ISS) is a collaborative programme secured via Intergovernmental Agreement between 15 governments and European Space Agency member states (10 European countries, UK, USA, Russia, Canada, and Japan) (ESA, n.d.). In 2007, the ISS Safety Task Force (IISTF) was tasked with delivering a report to the National Aeronautics and Space Administration (NASA) which reviewed safety controls and processes in place to avoid 3 high-risk scenarios that would impede ISS operations:

1. Destruction or abandonment of the ISS;
2. Loss of its crew;
3. Crew health issues (IISTF, 2007).

The report details no software solutions which directly address crew health. SecureSpace proposes development of a database management system (DBMS) application which allows ISS crew members to record health-related indicators to maintain a continuous overview of crew health throughout a mission. The system should keep track of crew radiation exposure, the effects of a closed/ unnatural environment with altered gravity, social isolation and Earth-distance, as primary hazards of space flight (Childress et al., 2023).

Domain-Specific Technical & Security Assumptions

- **Operating System (OS):** NASA systems run on Windows and Linux (The Linux Foundation, n.d.).
- **Approved cryptography & security:** NASA does not publish its in-force software security standards or cryptographic techniques. Standards published by OWASP, which encapsulate the most critical security risks and provide a consolidated framework for secure application development where otherwise no universal standards exist (Petranović & Žarić, 2023), will be referenced.

Proposed System & Requirements

While research into astronaut health monitoring is available (e.g. a Ballistocardiography (BCG) senor system (Kulau et al., 2022) or predictive diagnostics through vision testing in space (Fink et al., 2014)), SecureSpace found no evidence of a DBMS which delivers a continuous health record built on robust security architecture and strict data privacy principles.

The proposed application will allow astronauts to record health indicators related to physical fitness, mental health, ambient temperature and radiation exposure to monitor the risk of the crew developing health issues which may endanger the respective mission. The application will be available for use by astronauts visiting the ISS from all international partners (IPs) via an

onboard terminal. Astronauts will update their health data on an ongoing basis throughout a mission. To preserve privacy and data security, database records will be encrypted and deleted from the database following a mission.

Functional requirements:

- **Interface:** Command Line Interface (CLI), ensuring OS compatibility.
- **DBMS:** SQLite, selected for its stability, reliability, multiplicity and widespread use (W3schools, n.d.).
- **Downloads:** encrypted record downloads shall be allowed.
- **Concurrency:** limited to one user entering data and multiple sensors monitoring temperature, radiation etc.

Non-functional requirements:

- **Network:** Ethernet connection (100 mbps). The application will run on a central terminal to minimise attack surface and increase security of personal data (no onboard web access).
- **CPU:** the application will use threading for concurrency, not multiprocessing. It will therefore only require one processor core of a CPU, so it can be run on any machine.
- **RAM:** expected 1GB, typical of modern devices.
- **Storage:** Built-in memory. Estimated storage allowance for crew of 7 astronauts per mission (NASA, 2023) updating health records once daily is 20KB. Estimating 100 logs per health record per day at similar capacity, storage requirements for one year should not exceed a lower limit of ca. 722MB (2GB maximum). Logs will be stored in a separate database to health records.
- **Peripherals:** keyboard.

This proposal represents user-database interaction only. Transferring data to third parties (e.g. via email) is out of scope.

User roles will be limited to:

1. Superadmin (permissions: create user, assign roles and privileges, execute SQL queries for database management);
2. Mission Moderator (permissions: approve user, delete user profile and data);
3. Astronaut (permissions: update and view health records, download data files).

See Appendix 1 for UML diagrams:

1. Use case diagram, showing system scope;
2. Misuse case diagram, showing system security threats and safety hazards (Sindre, 2007);
3. Activity diagram, showing functional requirements;
4. Class diagram, showing system implementation as a basis for development.

Security standards

With the domain in mind, the application classifies as a level 3 standard verification (i.e. an application which delivers “high value, high assurance, or high safety”), referring to critical infrastructure or health and safety software (OWASP, 2021). Although OWASP (2021) advises application of the Application Security Verification Standard 4.0.3 (ASVS) in the Software Development Lifecycle (SDLC), this would be beyond the capacity of the team. Instead the widely-recognised OWASP Top 10 Application Security Risks (2017) are used as guidance. The following table summarises the vulnerabilities and security measures that will be implemented:

Security Risks	**Application Vulnerabilities & Planned Security Measures**
***A1:2017* Injection**	● Vulnerable injection surface: Superadmin database privileges (can perform SQL Queries). ● Security measures: Sanitize input (filter keywords and special characters) for all roles.
***A2:2017* Broken Authentication**	● Vulnerable authentication surface: user passwords. ● Security measures: multi-factor authentication; time-limited set-up passwords; password length and complexity requirements (min. 8/ max. 64 characters; allow most characters); limit, log and alert all failed login attempts.
***A3:2017* Sensitive Data Exposure**	● Data vulnerabilities: password storage; personal data. ● Security measures: encrypt database records; store passwords using strong salted hashing functions (bcrypt); penetration testing; delete sensitive data after mission.
***A4:2017* XML External Entities (XXE)**	● *Out of scope: application does not use XML parsers.*
***A5:2017* Broken Access Control**	● Authentication vulnerabilities: role-based access privileges for superadmin, admin and astronaut. ● Security measures: robust roles; no role-inheritance.
***A6:2017* Security Misconfiguration**	● Vulnerable security misconfiguration surface: external library/ database/ technology configuration and manipulation. ● Security measures: only implement necessary libraries; only implement OWASP-approved or known libraries; ensure correct configuration of all technologies.

A7:2017 **Cross-Site Scripting (XSS)**	• *Out of scope: application will not be available via a web browser.*
A8:2017 **Insecure Deserialization**	• *Out of scope: file downloads will be allowed, but deserialisation is not considered.*
A9:2017 **Using Components with Known Vulnerabilities**	• Potential component vulnerabilities: all implemented external libraries and database services. • Security measures: implement a tool that checks for dependencies and security vulnerabilities (bandit).
A10:2017 **Insufficient Logging & Monitoring**	• Loggable user actions: all 'auditable events' i.e. application errors; logins, viewing, updating and deletion of data; authentication successes and failures; authorisation failures; input/ output validation failures; excessive logins; memory changes etc. • Security measures: implement an effective monitoring and alert system for auditable actions; store log records in a dedicated database with restrictive commands; encrypt logs.

Table references: OWASP (2017); Pillai (2017); OWASP (2021) Cheat Sheet Series.

Tools and Libraries

The software will be written in Python and include the following compatible technologies:

- **Bcrypt**: for password hashing (recommended by OWASP (OWASP, 2021)).
- **Python built-in libraries**: re, logging, unittest, getpass, threading, pip.
- **Pylama**: covers pylint, pydocstyle, pycodestyle, pyflake, mccabe and others.
- **Bandit**: checks for common security flaws and dependencies.
- **Pytest Security**: checks for common issues such as input validation and access control.
- **coverage**: tests code coverage.
- **cryptography**: library for cryptographic algorithms.

The justifications for the use of these libraries are that they are supported in the most recent version of Python and developed through a rigorous process of proposing and accepting language changes.

GDPR Compliance

The domain dictates that the application may be used by ISS IPs, for whom data protection legislation is not universal. The application will therefore be developed according to the UK GDPR (Article 5(1), principally), which follows similar standards to the EU (ICO, n.d.). Article 9, 12 and 13 will apply as additional protections for data subjects in the collection of health data (i.e. consent and purpose of use) (Mulder, 2019).

The lawful basis for data processing is subject to:

1) the consent of the data subjects;
2) the vital interests of crew health as a key ISS vulnerability (IISTF, 2007);
3) the continuation of ISS operations, as outlined in the legal framework of the ISS (IISTF, 2007).

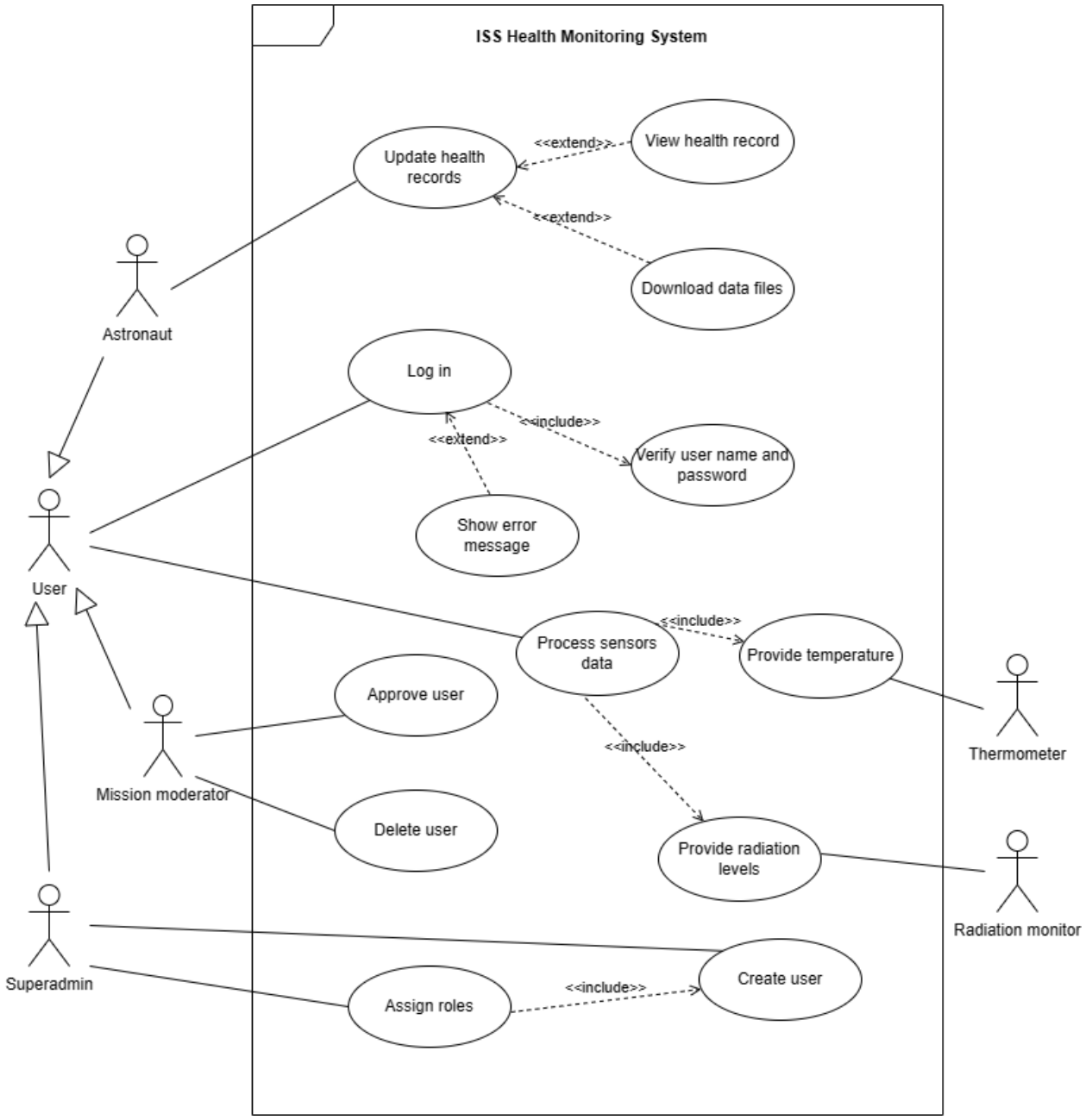
It is assumed that astronaut consent to data processing is sought before any mission commences. Nonetheless best practice dictates that data security, deletion and privacy statements/ policies are included in software documentation (alongside technical documentation) (Huth & Matthes, 2020). All personal data stored at rest within the DBMS will be encrypted, complying with data confidentiality principles (Ringman et al., 2018).

Testing

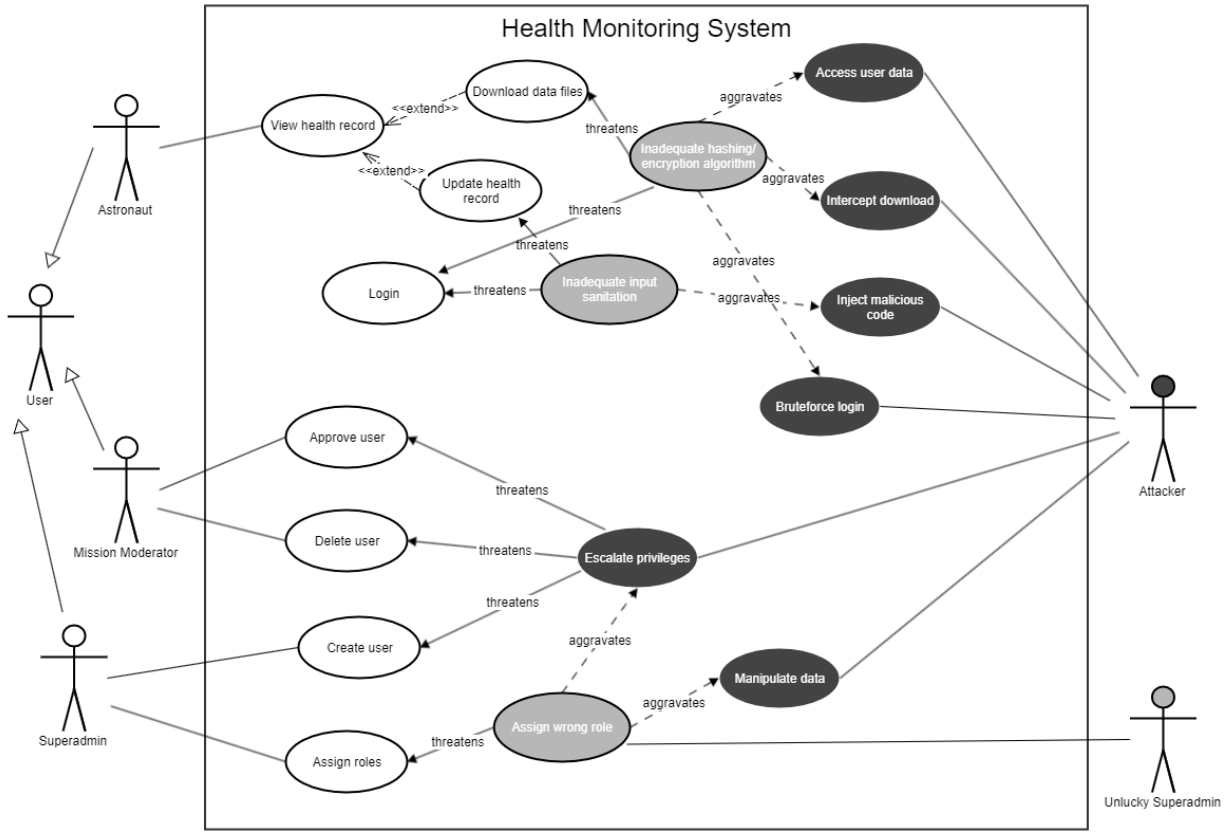
Whitebox testing is an ongoing process implemented by the development team itself (Pillai, 2017). Due to the small team size and limited timeframe, this is the most practical approach to testing. Unit testing (i.e. developing test cases for all functions and class methods and comparing the response with an expected outcome) will ensure that testing occurs systematically throughout the SDLC (Pillai, 2017).

Appendix 1: Application UML Diagrams

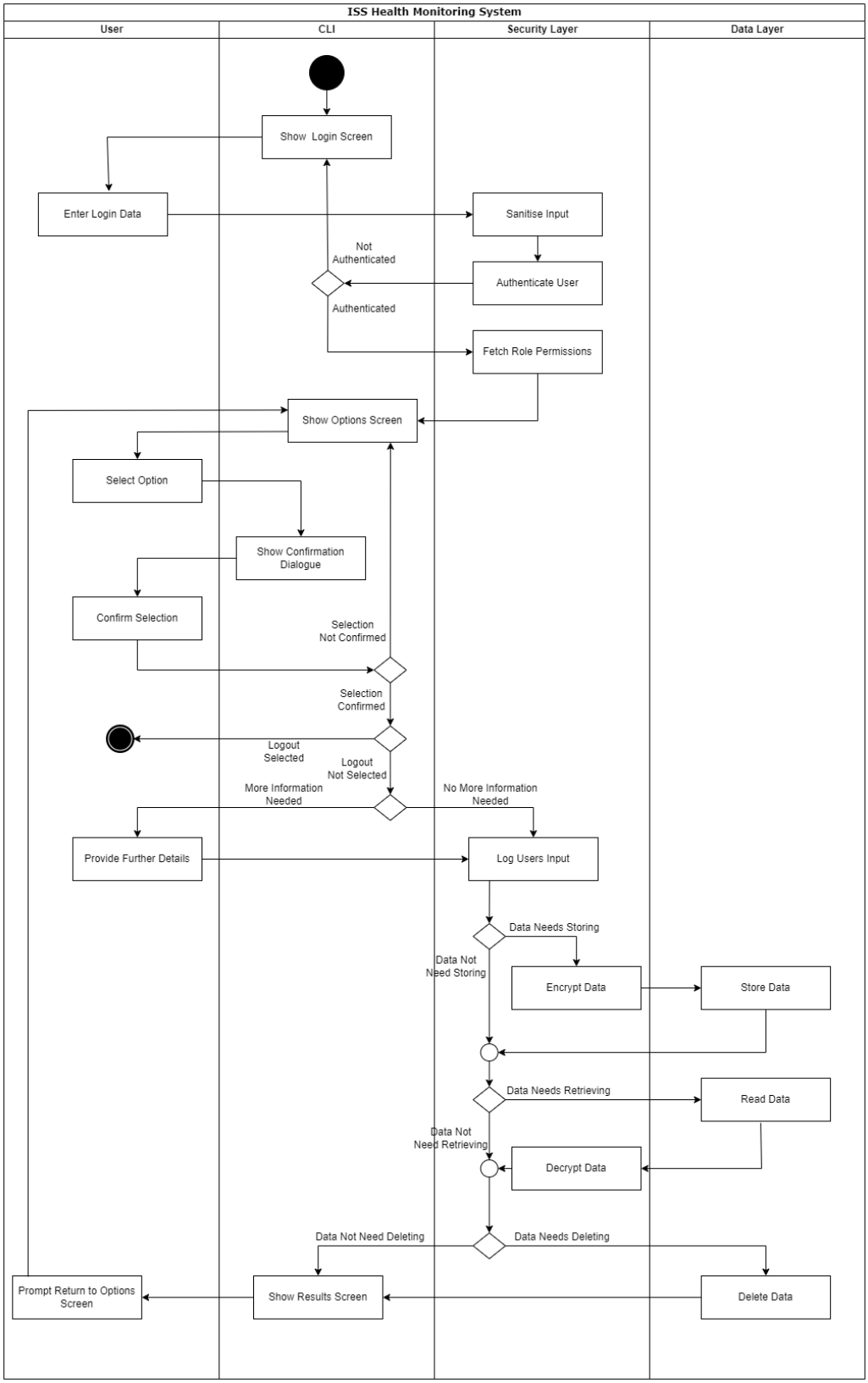
1. *Use Case diagram, showing system scope:*



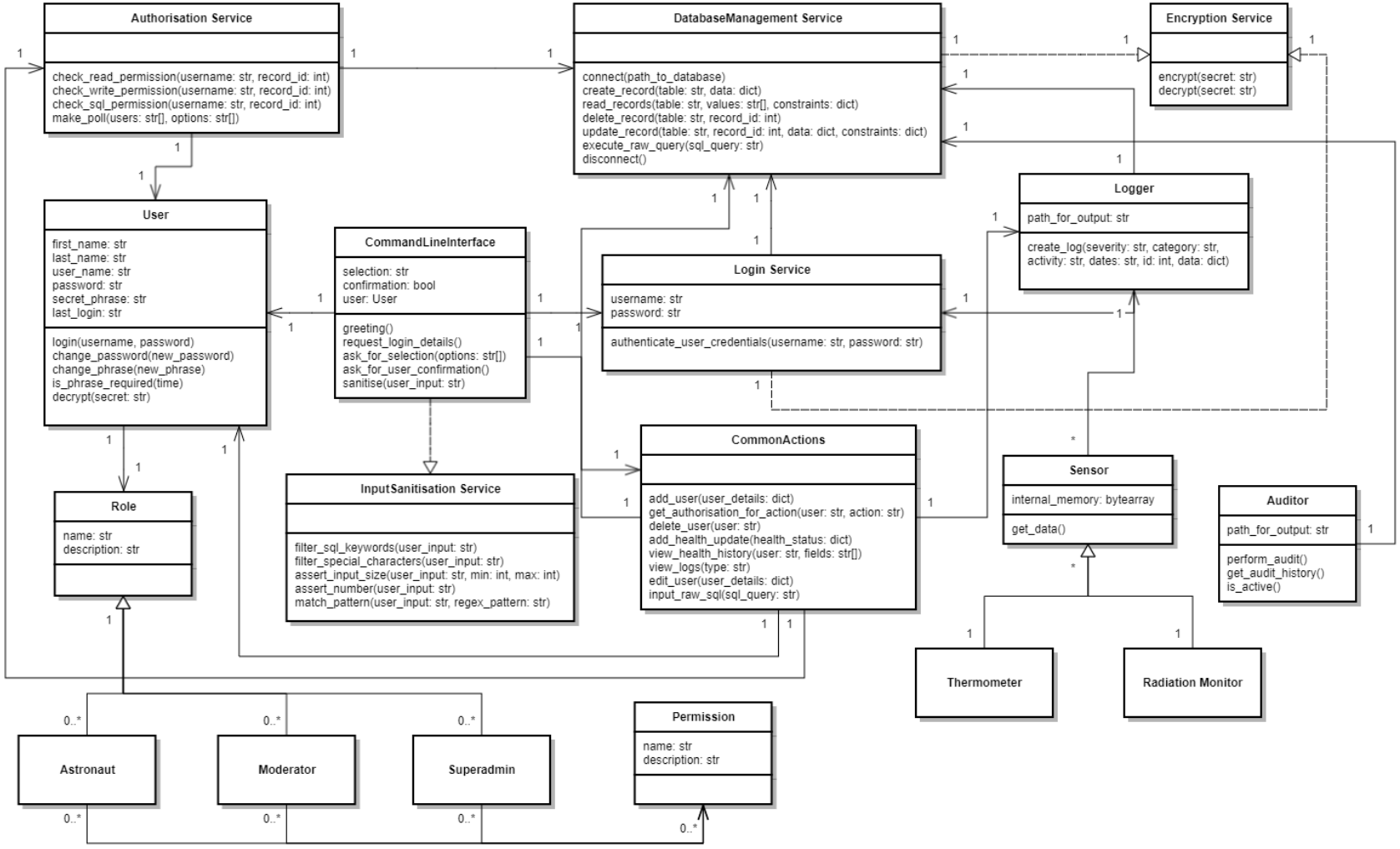
2. Misuse case diagram, showing system security threats and safety hazards (Sindre, 2007);



3. Activity diagram, showing functional requirements of the application:



4. Class diagram, showing system implementation as a basis for development:



References:

Childress, S.D., Williams, T.C. and Francisco, D.R. (2023) NASA Space Flight Human-System Standard: enabling human spaceflight missions by supporting astronaut health, safety, and performance. *npj Microgravity* 9(31). DOI: https://doi.org/10.1038/s41526-023-00275-2

ESA (n.d.) International Space Station legal framework. Available from: https://www.esa.int/Science_Exploration/Human_and_Robotic_Exploration/International_Space_Station/International_Space_Station_legal_framework#:~:text=The%20International%20Space%20Station%20is,Station%20in%20low%20Earth%20orbit. [Accessed 19 May 2023].

Fink, W., Popov, A. and Hess, A. (2014) Planning a pilot project on the ISS for crew health management & maintenance beyond LEO. *2014 IEEE Aerospace Conference*. Big Sky, MT, USA, 1-8 March. IEEE: Institute of Electrical and Electronics Engineers. 1-9. DOI: 10.1109/AERO.2014.6836505.

Huth, D. and Matthes, F. (2020). *ProPerData-A process model to support GDPR compliance*. Technical Report March, Technical University of Munich, Munich. DOI: 10.13140/RG.2.2.13926.37447

ICO (n.d.) Data protection and the EU in detail. Available from: https://ico.org.uk/for-organisations/data-protection-and-the-eu/data-protection-and-the-eu-in-detail/ [Accessed 31 May 2023].

IISTF (2007) Final Report of the International Space Station Independent Safety Task Force. NASA, pp 1-111.

Kulau, U., Rust, J., Szafranski, D., Drobczyk, M. and Albrecht, U. -V. (2022) 'A Differential BCG Sensor System for Long Term Health Monitoring Experiment on the ISS', *2022 18th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. Marina del Rey, Los Angeles, CA, USA. 30 May - 1 June. 85-92, DOI: 10.1109/DCOSS54816.2022.00028.

Mulder, T. (2019) Health Apps, their Privacy Policies and the GDPR. *European Journal of Law and Technology* 10(1). Available from: https://ejlt.org/index.php/ejlt/article/view/667 [Accessed 31 May 2023].

NASA (2023) International Space Station Facts and Figures. Available from: https://www.nasa.gov/feature/facts-and-figures [Accessed 4 June 2023].

OWASP (2021) *Application Security Verification Standard 4.0.3*. The OWASP Foundation. Available from: https://github.com/OWASP/ASVS/tree/v4.0.3#latest-stable-version---403 [Accessed 19 May 2023].

OWASP (2021) OWASP Cheat Sheet Series. Available from: https://cheatsheetseries.owasp.org/index.html [Accessed 31 May 2023].

OWASP (2017) OWASP Top 10 Application Security Risks - 2017. Available from: https://owasp.org/www-project-top-ten/2017/Top_10 [Accessed 19 May 2023].

Petranović T. and Žarić, N. (2023) 'Effectiveness of Using OWASP TOP 10 as AppSec Standard 2023', *27th International Conference on Information Technology (IT)*. Zabljak, Montenegro, 15-18 February. 1-4. doi: 10.1109/IT57431.2023.10078626.

Pillai A. B. (2017) *Software Architecture with Python* (Chapter 3, Chapter 6). Birmingham, UK. Packt Publishing.

Ringmann, S.D., Langweg, H. and Waldvogel, M. (2018) 'Requirements for legally compliant software based on the GDPR', *On the Move to Meaningful Internet Systems. OTM 2018 Conferences: Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018.* Valletta, Malta, 22-26 October. 258-276. DOI: 10.1007/978-3-030-02671-4_15.

Sindre, G. (2007) 'A Look at Misuse Cases for Safety Concerns', in: Ralyte, J., Brinkkemper, S., Henderson-Sellers, B. (eds) *Situational Method Engineering: Fundamentals and Experiences Proceedings of the IFIP WG 8.1 Working Conference, 12-14 September 2007, Geneva, Switzerland.* IFIP — The International Federation for Information Processing. 252-266.

The Linux Foundation (n.d.) Linux Foundation Training Prepares the International Space Station for Linux Migration. Available from: https://training.linuxfoundation.org/solutions/corporate-solutions/success-stories/linux-foundation-training-prepares-the-international-space-station-for-linux-migration/ [Accessed 31 May 2023].

W3schools (n.d.) Advantages of SQLite. Available from: https://www.w3schools.blog/advantages-sqlite [Accessed 26 May 2023].

Bibliography:

Al-Saqqa, S., Sawalha, S. & AbdelNabi, H. (2020) Agile Software Development: Methodologies and Trends. *International Journal of Interactive Mobile Technologies* 14(11): 246-270. DOI: https://doi.org/10.3991/ijim.v14i11.13269

Atlassian (N. D.) The Agile Coach. Available from: https://www.atlassian.com/agile#:~:text=The%20Agile%20methodology%20is%20a,planning%2C%20executing%2C%20and%20evaluating. [Accessed 12 June 2023].

ICO (n.d.) A guide to the data protection principles. Available from: https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/data-protection-principles/a-guide-to-the-data-protection-principles/ [Accessed 19 May 2023].

NASA (2010) NASA Extends the World Wide Web Out Into Space. Available from: https://www.nasa.gov/home/hqnews/2010/jan/HQ_M10-012_ISS_Web.html [Accessed 28 May 2023].

OMG. (2017) About the Unified Modelling Language. Specification Version 2.5.1. Available from: https://www.omg.org/spec/UML/2.5.1/About-UML#document-metadata [Accessed 11 June 2023].

Tudela, F. M., Bermejo Higuera, J.-R., Bermejo Higuera, J., Sicilia Montalvo, J.-A. and Argyros, M.I. (2020). On Combining Static, Dynamic and Interactive Analysis Security Testing Tools to Improve OWASP Top Ten Security Vulnerability Detection in Web Applications. Applied Sciences, [online] 10(24), 9119. DOI: https://doi.org/10.3390/app10249119.

WHO/EURO (2021) *The protection of personal data in health information systems – principles and processes for public health*. Copenhagen: WHO Regional Office for Europe.