CURRICULUM VITAE

Personal Information

Name: Gavrila (Martin) Adina Maria
E-mail: martin.adina92@gmail.com
Mobile Phone: +40756400376
LinkedIn Profile: Adina Maria Gavrila (Martin) | LinkedIn
Digital Resume: https://gavrilaadinamaria.github.io/Resume/

Education

2016 - 2018: Master's Degree in Embedded Systems, "Lucian Blaga" University, Sibiu, Romania, Faculty of Engineering - Hermann Oberth

2012 - 2016: Bachelor's Degree in Computer Science, "Lucian Blaga" University, Sibiu, Romania, Faculty of Engineering - Hermann Oberth

2008 - 2012: High School of Mathematics and Informatics "Grup Scolar Economic, Administrativ si de Servicii", Calimanesti, Rm. Valcea

Employment Career – Kendrion NV

November 2022 – present **Senior Software Engineer**

3 Interior Sound projects – Fisker AVAS Onebox, Seat A2B Interior sound and Audi SASGen29

- ➔ Created Python scripts for the sprint automation and workflow optimization.
- ➔ Build reliable, reusable and maintainable automated regression suites and test harnesses.
- ➔ Acts as a test lead, planning the work of other testing stuff and assigning tasks to meet the project deadlines.
- ➔ Design and develop automation solutions that meet the organization standards.
- ➔ Ensure best practices are followed and testing activities stay on schedule.
- ➔ Estimate and performing risk analysis for quality delivery.
- ➔ Capture quality assurance data and metrics to provide insights and conclusions.
- ➔ Participate in internal cross-team meetings, project scoping and functional review for assigned projects in an Agile environment.

- ➔ Review requirements and documentation to ensure that the individual requirements (functional and non-functional requirements) are testable.
- ➔ Review of the Software requirements.
- ➔ Review of the Software Sequence Design (sequence diagrams in Enterprise Architect) and analyze the corresponding source code of the involved interfaces.
- ➔ Create test specification for the software qualification tests, unit tests and integration tests.
- ➔ Create test environments for software unit test level and software integration test level using the VectorCAST tool.
- ➔ Implement and execute software unit and software integration test cases using Python and the VectorCAST tool.
- ➔ Prepare Python scripts for the report generation.
- ➔ Execute software qualification tests using the following main tools: CANoe, IAR Embedded Workbench,Ozone and J-link debugger.
- ➔ Summarize and communicate the results of the tests (including creation of the corresponding change request tickets).
- ➔ Create/Update Software Qualification/Unit/Integration Test Strategy / Test Plan and contribute to the overall test strategy together with the Test Manager.
- ➔ Participate to the review of the SW Architecture and create the relevant smoke tests.
- ➔ Support for the software deliveries to our testing teams and to our customer.
- ➔ Execute and analyze tests in Vector CANoe/VT System, logging results and identifying root causes of test failures.
- ➔ Create and maintain the CPU load tests and the BUS load tests.
- ➔ Perform sound measurement tests for amplitude and frequency (via oscilloscope, sound cards, Audacity tool, etc).
- ➔ Configure datasets for different functionalities: kickdown, one shot, virtual rpm, e-launch, etc.

Employment Career – ProIT Sibiu

March 2022 – November 2022 (8 months) Senior Software Test Engineer
2 Audi projects – DTC (Dynamic Torque Control) and DC(Dumper Control)

- ➔ Conducted TPT testing for model validation.
- ➔ Created Python scripts for process automation and workflow optimization.
- ➔ Performed dynamic testing (MIL/SIL/PIL) to validate the software functionality.
- ➔ Debugged MATLAB/Simulink models for issue resolution.
- ➔ Analyzed MATLAB-generated C code to enhance code coverage.
- ➔ Performed smoke tests and regression tests for each release from customer and bug fix releases.
- ➔ Reviewed requirements and documentation to ensure that the individual

requirements (functional and non-functional requirements) of the SWRS documents (module, block and software component) are testable and that the requirements do not contain any errors or ambiguities.

- ➔ Reviewed test specification (check also whether the test implementation adheres to guidelines and ensures continuous and advancing quality).
- ➔ Performed interface consistency check to ensure that modules of a SWC component are correctly integrated.
- ➔ Participated to the review sessions of different test implementation.
- ➔ Performed Integration Tests for each release.
- ➔ Black Box Testing for many modules (including Equivalence Class, Requirements Based and Boundary Values).
- ➔ Performed Resource Usage Test to ensure the fulfillment of resource requirements.
- ➔ Safety test obj files - testing on the target simulation board (ensure the correct behavior of the system under test with respect to the processor architecture e.g. size of scalar data types, error handling).
- ➔ Performed Back-to-back comparison (B2B) test between the model and the object on the target used to ensure that the behavior of the model with regard to the test objectives is equivalent to the automatically generated objects on the target (MIL - PIL comparison) within a specific tolerance.
- ➔ Performed parameter Validity Check (check that the dataset contains only desired parameter changes).

Employment Career – SII Romania

October 2021 – March 2022 Senior Software Integrator(extern) for HMI projects at Preh Brasov

2 projects – Daimler (Daimle_6425_HMI_SENSOR_LIN) and General Motors (GM_6346_HMI_DC_8in_Command)

- ➔ Created and maintained the Development Environment – BSG, Debugger Workspace, DDcom, Tessy, Polyspace – Bug finder and Code Prover, Bininspector, etc.
- ➔ Updated and described the tool usage in the Description of Development Environment document.
- ➔ Created and maintained the jobs for the Jenkins Server – fast, nightly and weekly jobs with particular commands.
- ➔ Created/Updated Software Integration Test Specification (+ Review together with the SW PM and SW Architect).
- ➔ Implemented Python Scripts for the automated testing.
- ➔ Created/Updated Software Integration Test Strategy.
- ➔ Created commands for the generation of different reports – DEV_SW_Testprotocol, SW_Documents_Protocol, SCM_report, etc.
- ➔ Software deliveries to our testing teams and to our customer.
- ➔ Release preparations (different integrations from mainline to branches

using Surround Tool).

- ➔ Updated and maintained the Release_Notes and the Release_Overview.
- ➔ Updated the linker file according to the specifications received from the SW Architect.
- ➔ Participated to the review of the SW Architecture and create the relevant smoke tests (for the SW Architecture entries that are constraints for the SwIntTS).
- ➔ Updated the analysis of different topics - using Test Track Tool.
- ➔ Daily interaction/discussions with the SW Developers, SW Architect and SW PM of the projects.

Employment Career – Continental Automotive Systems

March 2021 – October 2021 Software Development Engineer, Full Software Integrator and Changer Error Manager for a Daimler (MRA2) project (AUTOSAR project)

- ➔ Developer for Belt Hand Over – responsible for bug fixing and implementation of the new requirements (new DTC to inform the user about a malfunction of the belt hand over switch, new speed in case of some retraction/extension triggers).
- ➔ Software deliveries to our testing teams and to our customer - Daimler (at least one sprint per week).
- ➔ Integration of specific modules / features (Issues / Realization Orders, Change Packages) from mainline to dev path for 3 different variants of SW – MAX, MID2-3, MID1.
- ➔ Release preparations (configuration of the release script - update of the SW version, update of the data set version, etc.).
- ➔ Execution of the smoke tests.
- ➔ Cleaning of the release items – retarget the tasks that are not started /finished to the next sprint, create a list with all the integrated issues for our testing team, preparation of the vFlash packages, discussions with our testing team about the possible problems, etc.
- ➔ Daily interactions/discussions with the several members of the project: TPL, Change Error Manager, Scrum Master and developers from Sibiu, Singapore and Germany (Ingolstadt).
- ➔ Jenkins maintenance.
- ➔ CANoe maintenance.
- ➔ Execution and Maintenance for our Automatic tests (daily reports).

Dec 2019 – March 2021 Software Development Engineer for a Daimler (MRA2) project (AUTOSAR project)

- ➔ Development for adjustment modules - creation of a new module in Sibiu: Belt Hand Over module (belt bringer) and also many Diagnostics implementations including the following activities: requirements review, requirements clarifications, code implementation in IDEAS, bug fixes, analysis for different car traces and customer/testing issues, SW Detailed Design creation using Atom Tool, Unit Tests

creation/execution/automation, Software Tests with SWATT HIL, etc.

- ➔ Support / Bug Fixes / Integration for some old implementations (legacy code from MFA2 project = the base project for MRA2).
- ➔ Support/Backup for SW Integration.
- ➔ CANoe simulation generation via Model Generation Wizard tool.
- ➔ CANoe maintenance – panels creation, CAPL implementations to simulate different behaviors, support for the new Zenzefi versions and certificates, etc.
- ➔ Clarification/execution/automation/maintenance with SWATT HIL of different software specific tests (SWIT/SWT/SyIT) for the Functional Safety Module module (FSM).
- ➔ Unit tests creation for different modules (SWATT tests) – CAN, Manual Adjustment, Diagnostic.
- ➔ Smoke tests maintenance and automation – general smoke executed for every release.
- ➔ Support and Trainer for new colleagues – SWATT, IMS, IDEAS, CANoe, Code Collaborator.
- ➔ Integrator for an old Daimler project (MFA2) – 2/3 releases per year and support for the integrator of MRA2 project.
- ➔ Analysis for different customer(car) traces.
- ➔ Jenkins owner – maintenance of different scripts and jobs (QTools reports, automatic release, SWATT report, DEV Gate – continuous integration, nightly builds, etc.).

Jul 2018 – Dec 2019 Change Error Manager for Sibiu team and SW Integrator for Daimler(MRA2 and MFA2) projects (AUTOSAR projects)

- ➔ Performed SW integrations from mainline to different development paths.
- ➔ Jenkins slave maintenance – continuous(permanent) integration with Jenkins jobs.
- ➔ Release creation – createf binary files for every variant (MAX, MID2_3, MIN), QTools reports generation, SWATT Tests report generation, vFlash packs generation, checkpoints creation, execute SW Integration Tests/SW Tests/ Smoke Tests, report the results to many people involved in the project.
- ➔ Maintained SW Integration documentation.
- ➔ IMS Issues classification and planning for every release (handling of the tickets for the Sibiu, Germany and Singapore teams).

Jul 2017 - Jul 2018 Software Integrator for Daimler (MRA2) project (AUTOSAR project)

- ➔ Performed SW integrations from mainline to different development paths.
- ➔ Created and maintained SW integration plan and SW Integration strategy.
- ➔ Security Keys Integration – used for authentication.

- ➔ Release creation – created binary files for every variant (MAX, MID2_3, MIN), QTools reports generation, SWATT Tests report generation, vFlash packs generation, checkpoint creation.
- ➔ Created \ Executed Smoke Tests for every release (also create automation when possible).
- ➔ Created \ Executed Integration Tests for every release (also create automation when possible).
- ➔ Created \ Executed Software Tests for every release (also create automation when possible).
- ➔ Flash Bootloader (FBL) Integration and EOL Integration.
- ➔ Diagnostics implementations: mainly DTCs and DIDs.
- ➔ CANoe simulation generation via Model Generation Wizard tool.
- ➔ CANoe maintenance – panels creation, CAPL implementations to simulate different behaviors, Candela integration, etc.

Jun 2016 - Jul 2017 Software Developer for Ford C2 project (AUTOSAR project)

- ➔ Run Time Environment (RTE) – interfaces creation using Rhapsody and generation of the code using RTools.
- ➔ Helped for SW architects and developers – daily discussions and clarifications about the interfaces to be created.
- ➔ CANoe simulation generation via Model Generation Wizard tool.
- ➔ CANoe maintenance – panels creation, CAPL implementations to simulate different behaviors, Candela integration.
- ➔ Diagnostic implementation for Diagnostic Trouble Codes (DTCs) and Diagnostic Identifier Data (DIDs).

Jun 2014 - Jul 2016 Intern, Junior engineer, System Test team (student, part time – 4h per day)

- ➔ Tested BCMs - Body Control Modules for Porsche and Audi projects) – assigned modules: trunk lights and Controller Area Network (CAN).
- ➔ Requirements clarification and manual tests using CANoe.

Methodologies

- Software engineering
- AUTOSAR (COM, DIAG, RTE)
- V-Model Methodology
- Agile Scrum Methodology (involved in the full life cycle of the project, attended daily scrum meetings, sprint planning meetings and sprint review meetings)
- ASPICE key user (SWE.4 Software Unit Verification, SWE.5 Software Integration and Integration Test, SWE.6 Software Qualification Test, SUP.10 Change Request Management)

Programming Languages

- Embedded C /C

- Python (including Pytest)
- CAPL
- SQL (SQLite and PostgreSQL)
- C++
- MATLAB
- JavaScript
- TypeScript

Development and Testing Tools

- Eclipse (IDEAS)
- Microsoft Visual Studio
- CANoe
- DOORS
- Rhapsody
- IMS, MKS
- EB tresos Studio (Elektrobit, used for BSW configurations)
- Jira
- winIDEA
- Jenkins
- Surround
- Test Track
- Tessy
- Polyspace Bug Finder and Code Prover
- Binspector
- IAR Embedded Workbench
- Trace
- CodeCollaborator
- MATLAB Simulink
- TPT (Time Partition Testing)
- Toolcenter (internal automation tool used in ProIT)
- Enterprise Architect
- In-Step
- GIT
- J-link
- Vflash
- Vector CAST
- Turtoise SVN
- Enterprise Architect
- Ozone
- Postman (REST API Testing)
- Postman CLI
- Newman
- Continuous Integration and Continuous Delivery (CI/CD)
- Cypress
- Behavior-Driven Development (BDD)
- Chai
- Mocha
- Cucumber
- Selenium Webdriver

- Playwright
- JMETER
- Audacity
- VT System

Skills

- Team working skills
- Communication skills
- Organizational skills
- Patience
- Positive attitude
- Time management
- Precise on working schedules
- Curiosity to keep learning
- Creativity to bring fresh ideas to the table
- Proactive

Languages

- Romanian - native
- English - advanced (listening, speaking, reading, and writing)
- German - beginner

Trainings and certifications

- Problem solving and Decision Making (Know, March 2019)
- Agile Methodology (Colors in projects, 25-28 Aug 2020)
- AUTOSAR Training (Continental, 19-28 April 2021)
- Virtual Ownership Workshop (Continental, 8-9 July 2021)
- Training safety critical embedded system (Method Park, 16 -18 Aug 2021)
- ISQTB Foundation Level – ID: 22-CTFL-209668-14 (ISTQB® - International Software Testing Qualifications Board, June 2022)
- ISTQB® Certified Tester - Advanced Level - Test Analyst – ID: 24-CTAL-TA-254430-12 (ISTQB® - International Software Testing Qualifications Board, November 2024)

Courses

- Postman: The Complete Guide - REST API Testing – ID: UC-7d8e821f-8214-4b34-b05d-9fbdbc9d72f9 (Udemy, October 2024, https://udemy-certificate.s3.amazonaws.com/pdf/UC-7d8e821f-8214-4b34-b05d-9fbdbc9d72f9.pdf)
- Cypress -Modern Automation Testing from Scratch + Frameworks – ID: UC-ffe12691-c8f9-40cf-84c4-c2bc52652103 (Udemy, November 2024, https://www.udemy.com/certificate/UC-ffe12691-c8f9-40cf-84c4-c2bc52652103/)
- Selenium Webdriver with PYTHON from Scratch + Frameworks – ID : UC-a60c45eb-d637-4bc1-83b5-cb5552ccbc4d (Udemy, January 2025, https://www.udemy.com/certificate/UC-a60c45eb-d637-4bc1-83b5-cb5552ccbc4d/)
- Playwright test automation course with TypeScript. Best practices, API testing, Page Objects, Advanced features - ID: UC-c95b2069-8de8-4563-b374-94eca4d1eabf (Udemy,

March 2025, https://www.udemy.com/certificate/UC-c95b2069-8de8-4563-b374-94eca4d1eabf/)

- JMETER on Live Apps – Performance Testing – ID: UC-db735a8a-27e4-4594-b94b-ddcd053c1949 (Udemy, June 2025, https://www.udemy.com/certificate/UC-db735a8a-27e4-4594-b94b-ddcd053c1949/)
- Python/PostgreSQL – ID: UC-1ce71a0f-d5e9-499e-b4e9-d4c1aed4ab08 (Udemy, August 2025 - https://www.udemy.com/certificate/UC-1ce71a0f-d5e9-499e-b4e9-d4c1aed4ab08/)
- Development Programming Languages PySide Qt For Python (PySide6) GUI For Beginners : The Fundamentals (Udemy, October 2025 - https://www.udemy.com/certificate/UC-bc331549-2dd7-4d82-89ba-f679a0ddb2a0/)