

# Richard Louie Orilla

*Technical Lead of the  
Digital Shelf Analytics Team  
Senior Software Engineer*

Philippines | [richardorilla@pm.me](mailto:richardorilla@pm.me)  
[www.richardorilla.website](http://www.richardorilla.website)



<https://www.linkedin.com/in/richard-orilla/>  
<https://www.credly.com/users/richard-louie-orilla>  
[https://www.hackerrank.com/profile/hackerrank\\_clh7q](https://www.hackerrank.com/profile/hackerrank_clh7q)

## *About Me*

Dedicated problem solver and a researcher at heart. Kind of a person who likes to do challenging things for the sake of achievement.

Likes to explore whats new in tech, including dwelling on new programming languages. Favorite programming language is C and has already created real-time applications that is used for heavy-duty purposes using this language.

An avid fan of Linux yet explores other operating systems, including, BSD Family (Free, Open and Net), Haiku OS and even ICAROS (Amiga).

## *Educational Background*

*Masters of Science in Computer Science*  
[Incomplete]

De La Salle University (2017-2019)

*Bachelor of Science in Computer Science*

University of Mindanao (2010-2015)

## *Free and Opensource Software*

### *Contributor (Recent Examples)*

#### • *DosboxStagingReplacerForGogGalaxy*

- Created this project to replace the running Dosbox bundled on classic GOG games on GOG Galaxy to other Dosbox variants (especially Dosbox staging)
- Programmed this on modern C++ code because of the following:
  - Planned to make a lightweight CLI application with no unnecessary DLLs needed
  - Originally planned this on Rust but I instead use the former because I want to use the actual sqlite library code instead
  - I wanted to use modern coding features while also taking advantage of the original C sqlite code

#### • *WINE (Wine Is Not an Emulator)*

- Helped with bugfixes related to the portable executable port of ntdll.dll (as versus of in memory ntdll.dll) in order to make games like Street Fighter 5 work on Linux

#### • *Stratagus Engine - Wargus Branch*

- Helped debug an issue related to regression in music player on version 3.3.1 basically making the game unplayable with music on

#### • *OpenRA - Red Alert 2 Branch*

- Helped with several bugfixes with Red Alert 2 in OpenRA such as:
  - Incorrect speech notifications (when an enemy builds a super weapon)
  - Animation ticks fix (Some structures such as Allies Ore Refinery feel slower than the original)
  - Low powered animations (To be consistent with the original)

- *TediousJS (node-mssql)*
  - Implement feature that allows the library to support several Azure Active Directory connections

## *Work Experience*

- *Tech Lead - Analytics at inRiver (March 2018 - Present)*
  - Design and help in implementing several interconnected systems following the microservices architecture
    - Architectural Diagrams
    - Class Diagrams
    - Entity relationship Diagrams
    - Common libraries to be shared across different systems/projects (internal and external)
    - Assistance in implementation of CI/CD infrastructure
      - Creation of Dockerfiles
      - Creation of Helm charts
      - Creation of Azure Build Pipeline files
  - Several of these systems include:
    - Semantic analysis system that can determine what product and brand is a scraped title belongs to powered by machine learning and very complex, business centric decision trees
    - System that determines the performance of a product or brand on a website base on several business specific KPIs
    - Chromium browser based plugin that can scrape data in a website, including taking browser screenshots
  - Research several new technologies such as integrating LLMs to existing functionalities, analytics neural networks, computer vision (mainly using darknet) for business feasibility.
    - Created several detailed reports about the technology stacked
    - Created a neural network for classification purposes with the purpose of internally enhancing the PIM functionality
    - Created cost projection and made price adjustment suggestions as well (assuming such integration has low cost overhead and is still profitable)
    - Created proof of concepts of such technologies with non-stream line integration with the said existing functionality

## *Skillsets*

### *Programming Languages*

- |        |              |          |
|--------|--------------|----------|
| • C    | • Python     | • ASP    |
| • C++  | • Javascript | • PHP    |
| • C#   | • Ruby       | • Golang |
| • Java | • Rust       | • Kotlin |

### *Frameworks or Engine runtimes*

- |                    |                    |           |
|--------------------|--------------------|-----------|
| • Qt Framework     | • NodeJS + Express | • FastAPI |
| • Swing and JavaFX | • Ruby on Rails    |           |
| • Android SDK      | • Flask            |           |

## *Certifications*

- [Python Essentials 2 - Cisco](#)
- [Problem Solving Intermediate - Hackerrank](#)
- [Javascript Intermediate - Hackerrank](#)
- [HTML5 Application Development Fundamentals - Microsoft](#)