

Michael Rallo

Full Stack Software Engineer

CONTACT

☎ (314)-322-0042

✉ michael.s.rallo@gmail.com

in <http://linkedin.com/in/michael-rallo>

📍 2019 130TH PL SW, Unit B
Everett, WA 98204

Specialized Skills

Software Development:

OOPs – C++ & Java

GUI Tools QT (C++) & JavaFX (Java)

Low Level / OS – C & Cmake

Unit Testing (GTest)

Unreal Engine 4.0

Unity 5.X

OpenGL

Python

Web:

Web Services / REST APIs

Spring / Angular / React / Laravel

JavaScript / PHP

SQL

Dev Environment/Tools:

Git

Agile / SAFe

Pivotal Cloud Foundry / Jenkins

Virtual Machines

AWS / Azure

Team:

Communication

Design

Innovation & Motivation

Mentoring (Agile Processes / Software
Tools / Architecture)

EDUCATION

University of Missouri - Columbia

Bachelor's Degree

August 2014 - May 2017

➤ **Bachelor of Computer Science**

➤ **Majors:** Computer Science **Minors:** Mathematics

➤ **Latin Honors:** Cum Laude

EXPERIENCE

Boeing

July 2018 - Present

Software Engineer Grade II – Automation Integration

- Maintained/Updated Data, Metrics, and Applications used to display the statistics of the Automation performance using Tableau.
- Responsible for delivering valuable Software Applications to support automated manufacturing in the Fuselage Assembly Center. Java, Spring Boot, Spring Thyme, Maven, SQL Server were the usual toolsets utilized.
- Integrated Software onto the Fuselage Assembly Center, teaching engineers and managers how to use various tools I've engineered and developed. Examples include Activity Logging Tools and Equipment Issue Recording/Reporting tools which have reduced the need for mechanics and managers to spend time writing down / wandering through multiple Excel sheets – saving the company countless hours.

Boeing

June 2017 - June 2018

Software Engineer Grade I – Flight Simulation

- Responsible for defining the requirements; designing, developing, and testing software features; and supporting products through delivery and in the field.
- Developed in a Scaled Agile Framework alongside many other brilliant engineers.
- Daily C++ & Visual Studio Development which involved creating and maintaining features/tools for the Pilot Training Simulators and Instructor Operating System. Examples include generating/displaying map graphics and routes, GUI menus, application debugging, entity/object management, factory creation, DLL creation, etc.
- Distributed Interactive Simulation development in C++.

MBS

July 2016 - May 2017

Web Developer

- Responsible for independently Developing/Designing University Bookstore Websites from the ground up within an Internal Content Management System.
- Maintained Multiple Live Websites and provide excellent and proficient Client Services.
- Lead the company forward to pursue Responsive and Modern Web Designs.

Personal Projects

MadHouse – Unreal Engine 4.0

July 2018 – Present

Madhouse is a first person horror mystery and puzzle game created by a team of 3 artists and myself. Notable features I programmed are the inventory/crafting system, intractable interfaces when hovering/clicking on objects in the world, stress mechanics for the player, and AI for monsters throughout the game.

Budgie Adventures – Unreal Engine 4.0

September 2017 – November 2017

Budgie Adventures is a simple 2D side scroller platformer where the player controls a parakeet in a household environment.

Escape – Unity 5.X

March 2016 - May 2016

Escape is a solo first person adventure game in which the player has to escape from a dungeon filled with puzzles/traps.

Personal Projects – Cont.

Small Scaled Unreal Engine Projects

- 3D Pong.
- 3D Checkers.
- 2D Space Invaders.

University Projects

Dim3nsion

Dim3nsion was developed by a team of 6 for Capstone. This application takes an RGBD Image (Colored image with Depth), and produces an interactive 3D model. Dim3nsion utilizes a C++ library, PCL (Point Cloud Library), in order to extract data from a RGBD image and use the data to create a point cloud, which would then be used to create a 3D model. My contribution in this project was utilizing PCL to form a point cloud from which I used a triangulation algorithm to connect points in the cloud to form faces for the model to be rendered. The final product was a Web App from which the user can upload their image and receive a rotatable 3D object to download/view.

OpenGL – Object Loader/Viewer Engine

This Application was developed to process and view .OBJ files. Objects that are loaded in are scaled and translated to the origin. Users can view the object in Perspective or Orthogonal view. Translations and Rotations are also supported. Lighting can be adjusted and manipulated as well. Objects can be view by their points, vectors, or faces. Texture support was also implemented.

Vacuum AI

Vacuum AI is an application that is designed to send a virtual vacuum through a series of rooms in order to clean all rooms in an efficient manner. A StateSpace Map was used to determine which paths the AI should take. There are guidelines and costs for each action the vacuum takes. The AI utilizes the Iterative Deepening Search, the Depth First Search, and the A* Search to determine which moves to make. The point of this application was to demonstrate how different search algorithms effected performance of StateSpace prediction, which in this case was how fast a vacuum could clean a room.

Intersecting Line Segments

This OpenGL application visually displays the point(s) of intersection between two line segments, if any, stemming from two points in space. This was a small project to demonstrate how using projection points and the law of cosines can be utilized to find and visually portray the circle of intersecting spheres (such that a sphere represents the line segment's 360 degree rotation in all directions in space). The user may choose any two points and can navigate around in space freely.