

Jayant Bhardwaj

✉ hereisjayant@gmail.com | ☎ +1-604-404-2790 | 📍 5959 Student Union Blvd. Vancouver, B.C. V6T 1K2

Education

University of British Columbia (UBC)

BACHELOR OF APPLIED SCIENCE IN COMPUTER ENGINEERING

Vancouver

Sep 2018 – May, 2022 (expected)

Work and Volunteer Experience

Astin Analytics Pvt. Ltd.

DEVELOPER INTERN

Chandigarh, India

Aug 2020 – Present

- Implement a solution for mass thermal scanning with a team of developers
- Learnt about the OpenCV framework within the time constraints of the project
- Helped develop a server application, capable of handling multiple client requests simultaneously in Python for the project
- Successfully unit tested the client and server
- Coordination with customer's IT team

C.O.D.E. Hack

SENIOR TECHNICAL ADVISOR

Chandigarh, India

Jul 2020 – Aug 2020

- Ensured the smooth functioning of the C.O.D.E. Hack Hackathon
- Tested and evaluated the technology being used
- Trained the organizers with more efficient and economical techniques for evaluating the submissions
- Advocated for safer means of information exchange
- Assembled the architecture for the auto-graders of the event
- Provided training in git to the organizers

Alma Mater Society of UBC Vancouver

BARISTA AND CUSTOMER SERVICE REPRESENTATIVE

UBC, Vancouver

Jul 2019 – Mar 2020

- Process drink orders that range from basic to complex, quickly and efficiently
- Monitored inventory of supplies on hand
- Sell and merchandise other store products such as pastries, drinkware and seasonal items
- Take customer orders and process payments

UBC Advanced Research Computing (ARC) and Westgrid

VOLUNTEER FOR RESEARCH COMPUTING SUMMER SCHOOL

UBC, Vancouver

Jun 2019 – Jul 2019

- Assisted professors throughout their presentation as a TA
- Learnt about the possibilities that high performance computing and networking

Smarter.Codes

INTERN ANDROID DEVELOPER

Bangalore, India

Nov 2017 – Dec 2017

- Took inputs of voice packets from smartphones microphone, sent them over to AI API
- Parsed the outputs and presented the result to the user in form of text and speech
- Consume APIs and invoke them from a mobile application

Hamari Kaksha

ENGLISH AND MATH TEACHER (VOLUNTEER)

Chandigarh, India

Jan 2011 – June 2013

- Volunteered to teach English and Mathematics to a group of 24 underprivileged children

Technical Skills

Programming Languages: C, C++, Java, Python, JavaScript, HTML, Verilog, Assembly: x86-64, ARM

Framework and Tools: PyTorch, OpenCV, Android Studio, SolidWorks, MATLAB, Raspberry Pi, Arduino MCU
Git, Linux/UNIX, Windows, Pycharm, Google Colab, IntelliJ, ModelSim, Quartus

Technical Projects

Text classification using Sentiment Analysis (Jan-Feb 2021)

CLASSIFIED SHORT NEWS STORIES FROM A PRESS DATASET INTO FOUR CATEGORIES: WORLD, SPORTS, BUSINESS, AND SCI/TECH. TRAINED THE EMBEDDING VIA GRADIENT DESCENT

*Machine Learning, PyTorch,
Sentiment Analysis, Python, NLP,
Google Colab*

Image classification using Transfer Learning (Jan-Feb 2021)

USED TRANSFER LEARNING ON RESNET-18, BY REPLACING THE FULLY CONNECTED LAYER WITH ONE THAT HAS 2 OUTPUTS TO CLASSIFY THE CHARTER ARMS BULLDOG AND GLOCK 17, TRAINED THE NETWORK USING STOCHASTIC GRADIENT DESCENT THAT PRODUCED PERCENT ACCURACY IN THE HIGH NINTIES

*Machine Learning, PyTorch, Transfer
Learning, Python, Computer Vision,
Google Colab*

Big Integer Library and Memory Allocator for C (Jan-Feb 2020)

IMPLEMENTED DOUBLY-LINKED LISTS FROM SCRATCH IN C, AND USED IT TO IMPLEMENT THE BIG INTEGER LIBRARY. ALSO IMPLEMENTED A MEMORY ALLOCATOR FOR C, THAT USES AN IMPLICIT FREE LIST TO STORE BLOCKS THAT MAKE UP THE HEAP.

*C, Memory Allocator, Linux, Big
Integer, Valgrind, Memory
management, Computer Systems*

Debugger for y86-64 (Jan-Feb 2021)

IMPLEMENTED A GDB LIKE DEBUGGER FOR Y86-64 WRITTEN IN C AND Y86-64, SIMILAR TO WHAT GDB DOES FOR PROGRAMS WRITTEN IN C. IT SUPPORTS COMMANDS LIKE 'QUIT', 'STEP', 'RUN', 'NEXT', 'JUMP X', 'REGISTERS', 'BREAK X' ETC.

*y86-64, C, Debugger, Linux,
Computer Systems*

Mediator Service and Server for Wikipedia (Sep-Dec 2020) (UBC)

IMPLEMENTED AN FSFT(FINITE SPACE FINITE TIME) BUFFER THAT CAN HANDLE MULTIPLE THREADS TO BE USED AS DATA-CACHE. CREATE A MEDIATOR SERVICE FOR WIKIPEDIA USING THE JWIKI API THAT OBTAINS PAGES AND OTHER RELEVANT INFORMATION. THE MEDIATOR SERVICE SUPPORTS STRUCTURED QUERIES WHICH USE ANTLR FOR PARSER GENERATION. FSFT BUFFER WAS USED TO CACHE THE WIKIPEDIA PAGES TO MINIMIZE NETWORK ACCESSES. ALSO IMPLEMENTED A SERVER APPLICATION THAT WRAPS THE MEDIATOR SERVICE CAPABLE OF PROCESSING MULTIPLE CLIENT REQUESTS SIMULTANEOUSLY, USING JSON TO EXCHANGE INFORMATION.

*Java, FSFT Buffer, ANTLR, JSON,
JWiki API, IntelliJ*

Image Processing and Computer Vision Project (Sep-Dec 2020) (UBC)

IMPLEMENTED A DATATYPE USING CORE LIBRARIES IN JAVA THAT ALLOWS YOU TO PERFORM VARIOUS IMAGE PROCESSING AND COMPUTER VISION OPERATIONS ON AN IMAGE CALCULATING THE DFT, COSINE SIMILARITY OF THE IMAGES IN ORDER TO SUPPORT ADVANCED OPERATIONS LIKE GREEN-SCREENING AND TEXT-ALIGNMENT ALGORITHMS WITH COMPLETE TESTING USING JUNIT.

*Java, Image processing, JUnit,
Computer Vision, DFT, IntelliJ*

Analysing Social Networks (Sep-Dec 2019) (UBC)

IMPLEMENTED GRAPH USING ADJACENCY LIST AND ADJACENCY MATRIX AND GRAPH ALGORITHMS LIKE BFS AND DFS AND USED THESE ALGORITHMS IMPLEMENT FEATURES LIKE ANALYSING AN ANONYMIZED DATASET FROM TWITTER TO GET THE COMMON FOLLOWERS OF TWO USERS, AND THE MINIMUM NUMBER OF RETWEETS NEEDED TO GET USER A'S TWEET APPEARS IN USER B'S FEED.

*Java, Adjacency List, Adjacency
Matrix, BFS, DFS, Graphs, IntelliJ*

Interrupt Supported RISC Machine (Sep-Dec 2019) (UBC)

• DESIGNED AND IMPLEMENTED A 16-BIT INSTRUCTION SUPPORTED CPU ON THE FPGA BOARD DE1-SOC USING VERILOG, QUARTUS, AND MODELSIM. • I/O INTERRUPT SIGNALS VIA EXTERNAL KEYS AND SWITCHES WOULD CAUSE THE STATE MACHINE TO FINISH THE CURRENT INSTRUCTION, SAVE THE CONTENTS OF ITS REGISTERS, FLAGS AND PROGRAM COUNTER, RETRIEVE THE ADDRESS OF THE INTERRUPT SERVICE ROUTINE FROM A PREDEFINED LOCATION IN MEMORY, AND UPDATE THE PROGRAM COUNTER WITH THE ADDRESS.

*Verilog, Assembly: x86-64, ARM,
DE1-Soc, ModelSim & Quartus*