Karim Jabbour

✉ kareemjabbour@gmail.com   📞 438 229 8430   in linkedin.com/in/karimjabbour97   karim-jabbour.com

EDUCATION

McGill University		**Jan 2022 - May 2024**
BSc. Computer Science	GPA: **3.57/4.0**	Montreal, Canada
Concordia University		**Sep 2020 - Aug 2021**
BSc. Computer Science	GPA: **4.12/4.3**	Montreal, Canada

EXPERIENCE

Software Engineering Intern   **Flare Systems, Montreal**   **Sep. 2021 – Dec. 2021**
- Responsible for code base maintenance within Flare's own application, Firework, including bug fixes and code re-writing.
- Fixed UI issues such as wrong URLs, malfunctioning feeds, and broken buttons/links thus enhancing the user experience.
- Added several UI features to Firework like copy/paste button functionality, query-based highlighting, and new data feeds
- Automated admin login by loading necessary usernames/passwords of disconnected sessions from the backend (using Vue JS), allowing for a quick login without the need for copying any credentials.
- Designed and edited necessary tests for bug-fixes before running them through a GitLab CI/CD pipeline.

R&D Intern: AI Crypto Project   **AI Launch Lab, Montreal**   **Jun. 2021 – Aug. 2021**
- Processed the crypto-stock market dataset and extracted relevant financial indicators such as MACD, RSI, SMA, and EMA
- Applied classification algorithms on the processed data to categorize stocks as bearish, bullish, or neutral.
- Compared the efficiencies of different AI models (ex: Regression, LSTMs, and KNNs) in predicting future stock prices.

Research Assistant   **Prometheus Lab, McGill University**   **Sep. 2022 – Present**
- Developing new ways to implement hand gesture detection using existing computer vision libraries to allow for better accessibility features within modern applications.

Virtual TechEd Intern   **CEED, Concordia University**   **Jan. 2021 – Apr. 2021**
- Mentored a class of 30 middle-school students in Gulu, Uganda about the basics of computer use, Microsoft Office, and Python through online tech workshops and by writing them a visual handbook.

Extracurriculars

VP Finance   **McGill Syrian Students' Association**   **Sep. 2022 – Present**
- Handling the club's financial matters including audits, transaction lists, funding applications, and budget planning.

Head of Publicity   **Concordia Game Development Club, Montreal**   **Feb 2021 – Aug. 2021**
- Managed all the social media platforms for the club and wrote 5 articles for the club magazine

PROJECTS

COVID-19 detection through Chest X-rays *(McMed Hacks Hackathon Winner)*
- Designed a Convolutional Neural Network that classifies chest X-Rays as COVID-positive with an accuracy of around 90%.
  - *Tools: Python/Google Colab, KERAS, Matplotlib, NumPy, Pandas, Seaborn*

Ringtonify App
- Designed a ringtone maker using AWS services (Lambda, S3, Amplify) + React JS.
- Used AWS Lambda to host the main python code so that scaling and maintenance are done automatically.
- Setup Amazon S3 buckets to store the audio files and automatically clip them into ringtones.
- Designed a frontend UI using AWS Amplify that allows users to upload and download mp3 files to and from S3.
- Setup AWS Amplify on the frontend to force authentication through email.
  - *Languages: Python, React JS, Node JS, HTML, CSS*
  - *Tools: AWS Lambda, AWS S3, AWS IAM, AWS Amplify, AWS CLI, AWS Cloud-watch, VS Code*

Juke-Box Web App
- Built a Spotify-connected web app that allows multiple clients to search for and push songs to one hosting music player.
- Designed the front-end UIs for both Client and Host users such that:
  - Clients can easily access Spotify's database of songs without needing to authenticate their accounts.
  - Hosts can play the songs, display the next songs in queue, and show you the lyrics of the playing track.
- Designed the backend using Express.js and Mongo DB such that Spotify tracks can be pushed/popped from the database.
- Used an AWS EC2 instance to host the client-side web app on http://jukeboxapp.net.
  - *Languages: React JS, Node JS, HTML, CSS*
  - *Tools: Express, Mongo DB ATLAS, AWS EC2, NPM, Spotify-Web-API-Node, Lyrics-Finder API, POSTMAN, VS Code*

Daily Fortune App
- Designed a full-stack app (MERN) that sends daily fortune messages to clients through email.
- Built the backend using Docker, Python, and a MongoDB database that holds all the recipient emails.
  - *Languages: Python, React JS, Node JS, HTML, CSS*
  - *Tools: Express, Mongo DB, Mongoose, AWS EC2, NPM, DOCKER, SMTP, POSTMAN, VS Code*

SKILLS

- **Languages**: Java, Python, React JS, JavaScript, Node JS, C, Bash, GIT, NoSQL, HTML/CSS, Typescript, Assembly, AWK
- **Tools**: AWS (EC2, S3, Lambda, Amplify, CloudWatch), MongoDB, Unix/Linux, Express, NPM, GitHub, Postman, Docker