Avik Hasija

University of Waterloo
Systems Design Engineering

☎ (647) 772-0459

✉ a3hasija@uwaterloo.ca

🏠 avikhasija.com

AvikHasija

Skills

Languages

- Java
- C++
- Python
- JavaScript
- HTML/CSS
- XML

Tools, Frameworks, and Libraries

- Android SDK
- Crashlytics
- Volley
- Retrofit
- Butter Knife
- Node.js
- MongoDB
- SQLite

Awards

- **2nd Place** - Ontario Engineering Programming Competition, 2017
- **1st Place** - Waterloo Engineering Programming Competition, 2016

Experience

Native App Developer *CBC/Radio-Canada* *Jan '17 - Apr '17*

- Developed new features, interfaces, and bug fixes for major redesign of CBC News Android app in Java and XML, causing an increase in **MAUs of 16% to over 500k+**
- Implemented client-side functionality for **search, trending articles, and localized news** using internal CBC APIs
- Overhauled app navigation interface using Bottom Navigation components
- Worked with early developer preview of **Android Instant Apps** to create CBC News instant app split, which was unveiled at **Google I/O 2017**
- Used Media Sessions and Browser Services to develop a CBC News **Android Auto** MVP which streams CBC Radio programs

Lead Android Developer *Mindbend Studio* *June '15 - June '16*

- Worked on front-end design, user flow, and planning algorithmic operation of infor[me], an app to improve announcement systems in large organizations; released in closed beta to **200+ users**
- Built Android app in Java and XML with scalable organizational hierarchy and administrative post approval system
- Wrote **backend functionality** in JavaScript to subscribe users to organizations

Projects

Political News Bot

- Built Facebook Messenger chatbot in **Node.js and MongoDB** which performs **sentiment analysis** on articles using the Indico API to determine the news source's political stance
- User feedback is used to determine political affiliation; this model is continually adapted so users are served news aligned with their beliefs

Learning Lock

- Uses **Android services** and **AWS Machine Learning** to create a lock screen which can differentiate between owner and intruder
- Selected as **DevPost weekly staff pick** in February 2016
- Integrated lock pattern, pulled data from pattern nodes to train a neural network

PWR

- Power grid management simulator which visualizes path between power plants and homes, built with **JavaScript and Electron**
- Meets specifications by computing distance and financial metrics on an hourly basis
- Implemented **Djikstra's pathfinding algorithm** to reduce path inefficiencies

ITGS Prep

- Built card-based Android app following Material Design guidelines to help students study ITGS course content
- **500+ downloads** on the Google Play store with an **average 4.8 rating**