

Jakob Dubeau

Ottawa, ON, Canada

jakobdubeau.com | jakobdubeau@gmail.com | linkedin.com/in/jakobdubeau | github.com/jakobdubeau

EDUCATION

University of Ottawa

Bachelor of Computer Science

Ottawa, ON

Expected May 2027

TECHNICAL SKILLS

Languages: Java, Python, C#, JavaScript, TypeScript, HTML/CSS, SQL

Technologies/Frameworks: React, Next.js, Node.js, Three.js, TailwindCSS, Vite, Supabase, Firebase

Data/ML Tools: NumPy, Pandas, scikit-learn, Matplotlib, Seaborn, TensorFlow, Keras

Developer Tools: Git, VS Code, Visual Studio, Eclipse, Jupyter, Android Studio, Xcode, Vercel, Claude Code

PROJECTS

Discover Daily | *Next.js, React, JavaScript, TailwindCSS, Framer, Spotify Web API, OAuth 2.0*

- Built a **full-stack Next.js application** that generates Spotify playlists based on listening history, integrating frontend UI, backend APIs, and external services for authentication and business logic.
- Implemented **OAuth authentication with PKCE**, enabling secure user login and session handling with **HTTP-only cookies**.
- Built **REST API routes** to fetch personalized user data and generate playlist recommendations.
- Implemented **playlist generation logic** that builds 30-track playlists using a configurable **exploration-exploitation split**, removing duplicate tracks by ID and filtering songs with high recent play frequency.

End-to-End Machine Learning Pipeline | *Python, pandas, NumPy, scikit-learn, TensorFlow/Keras, Seaborn, Jupyter*

- Built an **end-to-end ML pipeline** in Python consisting of data auditing, preprocessing, model training, and evaluation, achieving ~ 0.86 **F1-score** on a held-out test set with consistent cross-validation performance.
- Designed **leakage-aware preprocessing and evaluation workflows**, enforcing strict train/test separation and k-fold cross-validation, resulting in $< 1\%$ discrepancy between validation and test metrics.
- Implemented and compared multiple classification models (Logistic Regression, KNN, Decision Trees, Random Forests), selecting a **Random Forest ensemble** that improved F1-score by approximately **10–14% over linear baselines** after hyperparameter tuning.
- Built and analyzed feed-forward neural networks using TensorFlow/Keras, detecting overfitting via learning curves and mitigating it with **early stopping, L2 regularization, and dropout**, restoring validation performance to within $\sim 1\%$ of training metrics.

Litematica Material List Visualizer | *Next.js, React, JavaScript, TailwindCSS, Framer*

- Built a **Next.js web application** that transforms Litematica .txt material lists into a Minecraft chest-style visual grid for improved readability and referencing over raw text output.
- Implemented a **robust parsing and validation pipeline** supporting multiple file formats, line-numbered ParseError reporting, and **safe client-side handling of uploads up to 10MB** with explicit retry and error states.
- Designed an **inventory normalization engine** that enforces Minecraft stack size rules, **groups and sorts items by category/quantity**, and paginates results into fixed 72-slot inventory grids.
- Built a scalable asset-resolution layer with **1300+ block/item texture mappings** and fallbacks, enabling consistent rendering and hover tooltips while maintaining smooth UI transitions via Framer Motion.

Online Tutoring Platform | *Java, Android SDK, Firebase, MVVM Architecture, Gradle*

- Designed a **role-based Android application** supporting Students, Tutors, and Administrators, modeling production workflows for authentication, approval-gated onboarding, scheduling, and session management.
- Applied **MVVM architecture** with repository abstractions to separate UI, business logic, and Firebase data access, **improving testability, modularity, and long-term maintainability**.
- Integrated **Firebase Authentication and Firestore** for secure user identity, persistent state, and transactional updates, including post-session tutor ratings and aggregated feedback.