

Michael M. Ashenafi

2715 Murray Avenue, Apt 320, Pittsburgh, PA, 15217 * (412) 419-6800 * michaelmogessie@cmu.edu

PROFESSIONAL EXPERIENCE

CARNEGIE MELLON UNIVERSITY, PITTSBURGH, PA

Educational Technology Researcher – Postdoctoral Fellow, May 2018 - Present

- Development and maintenance of educational technology software including cognitive tutors
- Analysis of educational data and development of predictive models of student performance
- Development of augmented reality solutions that support teachers in the classroom
- Testing the effectiveness of educational technology solutions by conducting school experiments
- Publishing research findings in journals and conference proceedings

Projects I led and was responsible for in this role:

- **Porting of legacy educational software built using Flash to current web standards** – I was tasked with implementing Flash-based education software that was used in schools for research purposes to a responsive web application that was capable of running across multiple web browsers and screen sizes. I led a team of four senior students from Carnegie Mellon University and the University of Pittsburgh to complete the task in three months. During that time, I introduced students to core Agile software development practices and methodologies including version control, collaborative documentation, sprint-based task planning, development and logging, bug reporting and code review.
 - Technologies, design methodologies and development practices I utilized in this project
 - Jira
 - Confluence
 - Bitbucket, HTML5, Javascript, CSS
 - JQuery
 - Twitter Bootstrap
 - **Development of Machine Learning Models that predict student affects** – The aim of this project was to develop data-driven models that predict student affects including confusion and frustration as they interact with education technology
 - Technologies I utilized in this project
 - Python – utilization of libraries including SciPy, scikit-learn, Pandas and Matplotlib for data analysis and visualization
 - Jupyter Notebook with Python - to explore and analyze data as well as test and build machine learning models
 - Machine Learning – classification algorithms such as ensemble methods, specifically XGBoost
 - **Development and maintenance of educational technology software using Augmented Reality** – This project was a collaboration between the Human-Computer Interaction Institute of Carnegie Mellon, the Robotics Institute of Carnegie Mellon University, and the Culinary Institute of America (CIA) in Hyde Park, New York. The aim of the project was to develop an AI-driven instructional technology prototype that would assist instructors at the CIA in monitoring the progress of trainee chefs in real-time as they perform activities in the kitchen. Concretely, this project would collect data through tactile, inertial, and three dimensional visual pose tracking sensors. The data gathered from the sensors would be analyzed in real-time to track and report the performance of individual students. The analyzed data would then be used to inform an instructor in the training kitchen of the status of the individual students through a visual
-

interface in the form of a smart-glass augmented reality (AR) device. The team chose to use the Microsoft HoloLens Augmented Reality Headsets for this project.

My roles and responsibilities in this project included:

- Development of a software back-end with a repository for storing student activity information and business logic for student data analytics
- Development of a RESTful web service API to receive information about student activity from the robotics team
- Development of a communication channel between the server and the Microsoft HoloLens headsets
- Visualization of student progress on the Microsoft HoloLens headsets

During the one year span of the project, I was able to develop a prototype of the proposed smart glass software as a client application that relied on updates from the back-end using a web socket implementation. I developed the back-end as a suite of RESTful web APIs that adopted a microservices architecture and made no assumptions about the specifications of clients that would consume those APIs.

- Technologies I utilized in this project
 - A **Spring Boot** Application to develop a **microservices** back-end with an Interface-Business-Repository pattern to decouple business and control logic and make software testing efficient and effective
 - **Spring Data** in combination with **Hibernate ORM** to deliver database access and implementation logic abstraction
 - **Spring Web Socket** and a **C# Web socket** client implementation that runs on the **Microsoft HoloLens Mixed Reality Headsets** to establish and manage communication between the smart-glass client and the back-end
 - **Unity 3D Virtual Reality Development** and **Microsoft Mixed Reality Toolkit**
 - **Google Firebase Cloud Messaging** for web browser notifications that can be subscribed to by any web-based client or a hybrid (or progressive web) application
 -
 - **Development and maintenance of a cloud-based service to aid educational technology research**
 - In this project, I was tasked with developing a system to address requirements that could not be met by a legacy system that is in operation. The requirements were for a system that was able to provide storage and retrieval services as well as intermediate-level analytics of student progress during an in class study. To address this issue, I developed a Spring Boot application that was simply a suite of RESTful applications and followed a microservices approach to developed loosely coupled services. The services was implemented as a dockerized application deployed in the cloud.
 - Technologies I utilized in this project
 - A **Spring Boot** Application to develop a **microservices** back-end with an Interface-Business-Repository pattern to decouple business and control logic and make software testing efficient and effective
 - **Spring Data** in combination with **Hibernate ORM** to deliver database access and implementation logic abstraction
 - **Docker** for containerization of the application
 - **Git** with **BitBucket** for version control
 - **Jenkins** for continuous integration using the **BlueOcean** plugin for creating a deployment pipeline
 - **NGINX** as an interface for routing HTTP requests to multiple dockerized applications running in the cloud
-

FIRLAB SRL, FLORENCE, ITALY

Software Engineer, Dec 2015 - Apr 2018

- Development and maintenance of an enterprise-grade human resource scheduling and management (Software as a Service) SaaS solution
- Development and maintenance of an API-driven hybrid mobile app client for the SaaS solution
- Software project management

Projects I led and was responsible for in this role:

- **Design, development, testing and analysis of algorithms for human resource scheduling and management for large enterprises** including hospitals, manufacturing plants, news agencies, airports and marine terminals – this project required designing heuristic-based algorithms that aimed to generate optimized work shifts for employees in accordance with rules provided by supervisors and managers while adhering to statewide and federal regulations.
 - Technologies, design methodologies and development practices I utilized in this project
 - **J2EE – Java Server Pages (JSP)** for building views and **servlets** for building controllers
 - **Interface-Business-Repository Design Pattern** to decouple business and repository logic to guarantee seamless software testing
 - **JUnit** and **H2** in-memory database for unit and integration testing to derive **Test-Driven Development**
 - The **Model-View-Controller (MVC)** pattern
 - **Hibernate Object Relational Mapping (ORM)** to abstract database access and implementation logic and ensure compatibility across multiple database vendors including **Oracle, Postgres, Microsoft SQL** and **MySQL**
 - **Spring MVC** framework built on-top of business and repository logic
 - Development of user interfaces using **Twitter Bootstrap, Javascript, JQuery, CSS3** and **HTML5** to deliver a highly interactive and responsive user experience in viewing and updating shifts and other schedules using interactions such as drag-and-drop, resizing and moving elements.
 - Creation and utilization of user interface templates to leverage the MVC design pattern and minimize client-side rendering logic duplication using the MVC framework **Backbone.js** and the Javascript templating language **Lodash (formerly Underscore)**
 - **Drools** Business Rules Management System (BRMS) to develop, test and implement business rules as dictated by company, state and federal regulations when generating and evaluating employee schedules.
 - **Docker** for building and deploying enterprise applications to data centers while ensuring a containerized environment that ensures compatibility, simple application environment configurations and rapid deployment.
 - **Jenkins** for continuous integration of new features and hot fixes as they are pushed to remote GIT repositories.
 - **Apache Camel** to implement communications between components of the application using a producer-consumer pattern – for instance, to notify the a requesting subsystem that the generation of an employee schedule is complete.
 - **Google Firebase Cloud Messaging (FCM)** to develop the back-end of a push notification service for employees utilizing enterprise mobile applications.
 - **SonarQube** to monitor code quality including bugs, codes smells and vulnerabilities

- **SOAP web services** to communicate with a Sonos Wireless Smart Speaker – In one of the projects I was heavily involved in, I was tasked with developing a software/hardware solution to fulfill a requirement by ground air traffic personnel to notify Air Traffic Control in real-time when an aircraft was ready to depart after successful inspection. Consequently, I led my team in developing a solution that leveraged web sockets and SOAP web services to notify Air Traffic Control personnel via the Sonos Wireless Smart Speaker as soon as a flight was ready to depart.
 - **Jersey RESTful Web Services framework** using **Jackson** to develop JSON-based API endpoints that provide interfaces for mobile applications and other clients.
 - **JSON Web Tokens (JWT)** for authenticating web service API calls
 - **Design, development, testing and deployment of a real-time web application for efficient and enhanced communication between ground air traffic personnel and Air Traffic Control**
 - Technologies, design methodologies and development practices I utilized in this project
 - **Atmosphere** – A web socket implementation in Javascript for the client-side and in Java for the back-end
 - **Jquery mobile** – for a mobile-first design approach
 - **Backbone.js** – to adopt an MVC framework
 - **Lodash (formerly Underscore)** for Javascript templating in order to take advantage of code reuse and improve rendering efficiency
 - **JQuery Ajax** to communicate with RESTful web APIs
 - **Selenium Automation Testing** to test the functionality of the application's Graphical User Interfaces (GUI)
 - **PouchDB and Local Storage** for storing temporary user data inside a Web Browser
 - **Design, development, testing and deployment and migration of a hybrid mobile applications for employees of clients**
 - As part of a small company, I was tasked with the sole responsibility of developing a mobile application employees of clients would use to explore their weekly, bi-weekly or monthly work shifts, request leaves and absences and submit shift preferences. The requirements included messaging capabilities among employees, in accordance with company rules, and with supervisors. Employees also had an option to opt in and out of receiving specific notifications from the back-end system. Due to the small size of my company, my team decided to pursue the development of a hybrid mobile application that would have a native look across platforms and would be capable of being deployed on multiple platforms with minimal code modification or manipulations. I was able to develop a hybrid mobile application that met all the specifications, implemented minimalist business logic and was heavily tasked with rendering user interfaces. This approach allowed the development of a mobile application that was largely business logic-agnostic and relied on the back-end for most rendering logic. Ultimately, this allowed us to develop an application that could be highly customized as needed using back-end logic without requiring users to either upgrade or install/uninstall the current instance of the application. Updates to user interfaces and access to features were enforced in real-time using cloud messaging features that were driven by changes to Access Control Logic (ACL) resulting from updates from departmental policies and/or company regulations. Based on clients' requirements, additional features of our mobile application needed to include the ability for employees to clock-in and clock-out without using tactile technologies. Accordingly, we developed a timekeeping module as a component of our application using Low-Energy Bluetooth (BLE) technology.
 - Technologies, design methodologies, hardware and development practices I utilized in this project
 - **AngularJS** framework for hybrid mobile logic development to
 - **Ionic v1** for User Interface (UI) development
 - **Google Firebase Cloud Messaging** to receive push notifications from the back-end
-

- **The Raspberry PI mini-computer** to act as a clocking terminal that interfaces with an employee's smartphone via the app using Low-Energy Bluetooth (BLE) technology
- **Node.js** to develop a server that runs on the Raspberry PI via BLE and relays messages to the backend as clock-in and clock-out messages are received from employees' smartphones
- **JSON Web Token (JWT)** strings that are embedded in HTTP Auth headers for server-side authentication
- **Cordova Platform** for compiling web apps into platform-specific apps and for taking advantage of community plugins that provide native application capabilities such as push notification, device permissions, storage, camera, location services and other platform-specific features.
- **Migration of the current mobile application from AngularJS/Ionic v1 to Angular 7/Ionic 4**
 - A year after the completion of the mobile application using AngularJS/Ionic v1, we as a team were convinced that this framework was too old to maintain and take advantage of new features. We were faced with either migrating to the latest version of AngularJS and Ionic, or re-implementing the application from scratch using the new frameworks. Given the major re-implementations of both frameworks and how costly, time-intensive and that fact that it would eventually require maintaining a legacy portion of the software, we decided to re-implement the application using Angular 7 and Ionic 4. I was tasked with the sole responsibility of this task and successfully delivered the application as a part-time employee in October 2019.
 - New technologies, design methodologies, hardware and development practices I utilized in this project
 - **Angular 7 (TypeScript)**
 - **Ionic 4**
 - Technologies, Platforms and Software Development Practices across all projects
 - **Agile Software Development** Practices – Sprint planning, bug reporting, timely task completion, stand up meetings, task creation, progress logging, code review using **Jira**
 - Collaborative documentation of projects, procedures, knowledgebase and user requirements using **Confluence**
 - Pair programming
 - Version Control using **Git** – Bitbucket (formerly Stash)

FINOTE TIBEB SCHOOL, ADDIS ABABA, ETHIOPIA

Secondary education teaching professional, Sep 2006 – Sep 2010

- High school instructor for Critical Thinking and Reasoning Skill courses
- High school instructor for introductory course in Information Technology

PERSONAL AND DEMO PROJECTS

- **A Taxi data RESTful web service architecture** - a cloud-hosted data service to ingest, store and analyze data.
 - Technologies, Platforms and Software Development Practices across all projects
 - Version Control using **Git**
 - The source code is hosted on **BitBucket Cloud** and is publicly accessible at <https://bitbucket.org/michaelmogessie/data-service/>
 - The data service is deployed on a **Compute Engine** virtual machine on **Google Cloud Platform**.
 - The entire data service is built as a docker image using maven and deployed as a containerized application. The database is also a **dockerized MySQL** container.
-

- A **NGINX** server provides port forwarding to provide access to the data service container and WWW access to a directory containing the demo web page.
- The service deployment uses **Jenkins** to automate application delivery in a **Continuous Integration (CI)** manner.
- The deployment pipeline, which was created using the Blue Ocean plugin, checks out the source code from the master branch on BitBucket and builds and deploys the image.

EDUCATION

UNIVERSITY OF TRENTO

Trento, Italy

Ph.D. Information Technology (Nov 2017)

- **Dissertation:** An Online Peer-Assessment Methodology for Improved Student Engagement and Early Intervention

UNIVERSITY OF TRENTO

Trento, Italy

M.S. Computer Science (Sep 2012)

- **Relevant Courses:** Object-Oriented Software Design, Web Architectures, Concurrent Programming, Embedded Systems, Advanced Networking, Security for IT

ADDIS ABABA UNIVERSITY

Addis Ababa, Ethiopia

B.S. Information Systems (Sep 2006)

- **Relevant Courses:** System Analysis and Design, Object-Oriented Software Design, Information System Project Management

CERTIFICATES

- Advanced Machine Learning with TensorFlow on Google Cloud Platform – Google Cloud (November 2018)
- Deep Learning Specialization - deeplearning.ai (December 2018)
- Applied Data Science with Python Specialization - University of Michigan (October 2018)

ADDITIONAL SKILLS, AND EXPERIENCES

- Machine Learning and Deep Learning - Classification, Regression, Clustering – Convolutional Neural Networks and Recurrent Neural Networks with TensorFlow, Keras, Natural Language Processing (NLP)
 - Unix Operating System – CLI and Bash, Cron scheduling, HTTPS certificate signing request (CSR) generation, submission and certificate installation in Node.js and NGINX
 - Apache Solr
 - Google Cloud Platform, Google Colab
 - TCP/IPv4, UDP, FTP, SSH, HTTP, HTTPS, POP, IMAP, SMTP
 - SQL, JSON, XML, XSD, AIML
 - Ubuntu, Mac OS X, Windows
 - Excellent verbal and oral communication skills – I have lived in three continents and worked in diverse cultures and across multiple demographics. Through years of experience as a student, an apprentice, a teacher and a professional, I have acquired invaluable skills in communicating with clients, colleagues, upper-level management and interns, both in formal and informal settings. I speak three languages (Amharic, English, Italian) and have lived in three continents, so far.
-

PUBLICATIONS

M. Mogessie, S. D. Wolf, M. Barbosa, N. Jones and B. M. McLaren, "Work-in-Progress—A Generalizable Virtual Reality Training and Intelligent Tutor for Additive Manufacturing," 2020 6th International Conference of the Immersive Learning Research Network (iLRN), San Luis Obispo, CA, USA, 2020, pp. 355-358, doi: 10.23919/iLRN47897.2020.

Mogessie M., Elizabeth Richey J., McLaren B.M., Andres-Bray J.M.L., Baker R.S. (2020) Confusion and Gaming While Learning with Erroneous Examples in a Decimals Game. In: Bittencourt I., Cukurova M., Muldner K., Luckin R., Millán E. (eds) Artificial Intelligence in Education. AIED 2020. Lecture Notes in Computer Science, vol 12164. Springer, Cham. <https://doi.org/10.1007/978-3->

Ashenafi, M. M. (2019). Online Peer-Assessment Datasets. arXiv preprint arXiv:1912.13050.

Richey, J. E., McLaren, B. M., Andres-Bray, M., Ashenafi, M., Scruggs, R., Baker, R., & Star, J. (2019). Confusion in Learning from Erroneous Examples: Does Type of Prompted Self-explanation Make a Difference?. In International Conference on Artificial Intelligence in Education (pp. 445-457). Springer, Cham.

Richey, J. E., Andres-Bray, J. M. L., Ashenafi, M., Scruggs, R., Andres, J. M., Star, J. R., ... & McLaren, B. M. (2019). More confusion and frustration, better learning: The impact of erroneous examples. *Computers & Education*, 139, 173-190.

Ashenafi, M.M. A Comparative Analysis of Selected Studies in Student Performance Prediction. 2017. *International Journal of Data Mining & Knowledge Management Process*.

Ashenafi, M. M., Ronchetti, M., & Riccardi, G. (2016, October). Exploring the Role of Peer-Assessment as a Tool of Early Intervention. In ICWL 2016 - 15th International Conference on Web-based Learning.

Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2016, November). Using Students' Collaboration to Improve Active Participation in University Courses with Large Number of Attendees. In 9th annual International Conference of Education, Research and Innovation.

Ashenafi, M. M., Ronchetti, M., & Riccardi, G. (2016, July). Predicting Student Progress from Peer-Assessment Data. In 9sup th International Conference on Educational Data Mining, International Educational Data Mining Society.

Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2015, October). Predicting students' final exam scores from their course activities. In *Frontiers in Education Conference (FIE)*, 2015. 32614 2015. IEEE (pp. 1-9). IEEE.

Ashenafi, M. M. (2015). Peer-assessment in higher education—twenty-first century practices, challenges and the way forward. *Assessment & Evaluation in Higher Education*, 1-26.

Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2014, June). A Web-Based Peer Interaction Framework for Improved Assessment and Supervision of Students. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications* (No. 1, pp. 1371-1380).

Ashenafi, M. M., & Ronchetti, M. 2012. Lode4Android: bringing rich video-lectures into an app. In World Conference on Educational Multimedia, Hypermedia and Telecommunications (Vol. 2012, No. 1, pp. 2921-2928).
