From Students…
…to Professionals

MICHIGAN STATE UNIVERSITY

Project Plan
Transaction Anomaly Detection

The Capstone Experience

Team MSUFCU

Austin Roberts
Andrew Schmidt
Caleb Sherman
Paul Soma
Jiaming Xu

Department of Computer Science and Engineering
Michigan State University
Fall 2018
Functional Specifications

• Detect and alert users of anomalous transactions
  ▪ Fraud / account takeover
  ▪ Changes in recurring payments

• Provide visualization of member spending habits

• Implement anti-money laundering rules according to Bank Secrecy Act
  ▪ Stretch goal
Design Specifications

• Create web, Android, and iOS applications for user interaction
• Use colors and icons to indicate severity of anomaly
• Potentially fraudulent transactions will be brought to an employee’s attention for review
• Members will be alerted according to MSUFCU’s notification protocol
Screen Mockup: Mobile App
Screen Mockup: Member Web Portal
Screen Mockup: Member Web Portal
Screen Mockup: Employee Web Portal
Technical Specifications

• Server: Amazon EC2, AWS Elastic Beanstalk
• API: Django REST Framework
• Mobile Apps: iOS, Android
• Web Portals: Member-facing, employee-facing
• Database: MySQL hosted on AWS
System Architecture

- **Amazon Web Services**
- **MySQL** (Database)
- **TensorFlow**
- **Python**
- **Keras**
- **Anomaly Detection Module**
- **Django** (Web server)
- **RESTful API**
- **HTML GET Request JSON Response**
- **Android**
- **iOS**
- **Member Web Portal**
- **Employee Web Portal**
System Components

• Hardware Platforms
  ▪ All servers on Amazon Web Services

• Software Platforms
  ▪ Member-facing iOS, Android, Web platforms
  ▪ Employee-facing Web platform

• Technologies
  ▪ Scikit-learn
  ▪ TensorFlow / Keras
  ▪ Django
Risks

- Quality of data
  - Dataset has very few instances of fraudulent transactions
  - Mitigation: Techniques such as local outlier factor and single class support vector machines are designed to deal with this issue

- Connecting all components
  - iOS and Android apps are not connected to the AWS database
  - Have identified tutorials and resources on how to connect apps to AWS

- Provide value to members
  - Don’t know how best to deliver meaningful information to members
  - Will utilize focus groups to test our apps and will implement changes based off of their feedback

- Security
  - Must ensure transaction data is secure from end to end
  - Mitigation: Having no memory leaks and isolating our modules
Questions?