MICHIGAN STATE UNIVERSITY

Project Plan Presentation

"Spaving": Giving based on Spending Habits

The Capstone Experience

Team MSUFCU

Jonathon Harkness
Ning Wang
Ethan Colbert
Nick Aaltonen

Department of Computer Science and Engineering
Michigan State University

Fall 2021

From Students…
…to Professionals

Functional Specifications

- Better recommendation system for the spending analysis page
- Notifications to the user alerting them of new recommendations
- Detailed insights and information regarding the recommendation

Design Specifications

- Use machine learning to generate various recommendations for the user
- Design a dedicated recommendation page to display generated list of potential charities and spending comparisons
- Display various different notifications depending on the type and quantity of the recommendations

Screen Mockup: Screen Notification

Figure 1: Multi-Charities Notification

Figure 2: Single Charity Notification

Screen Mockup: Recommendation

Figure 3: Recommendation Button

Figure 4: Detailed Page of Recommendation

Figure 5: Recommendations page with swipe

Screen Mockup: Charity Information

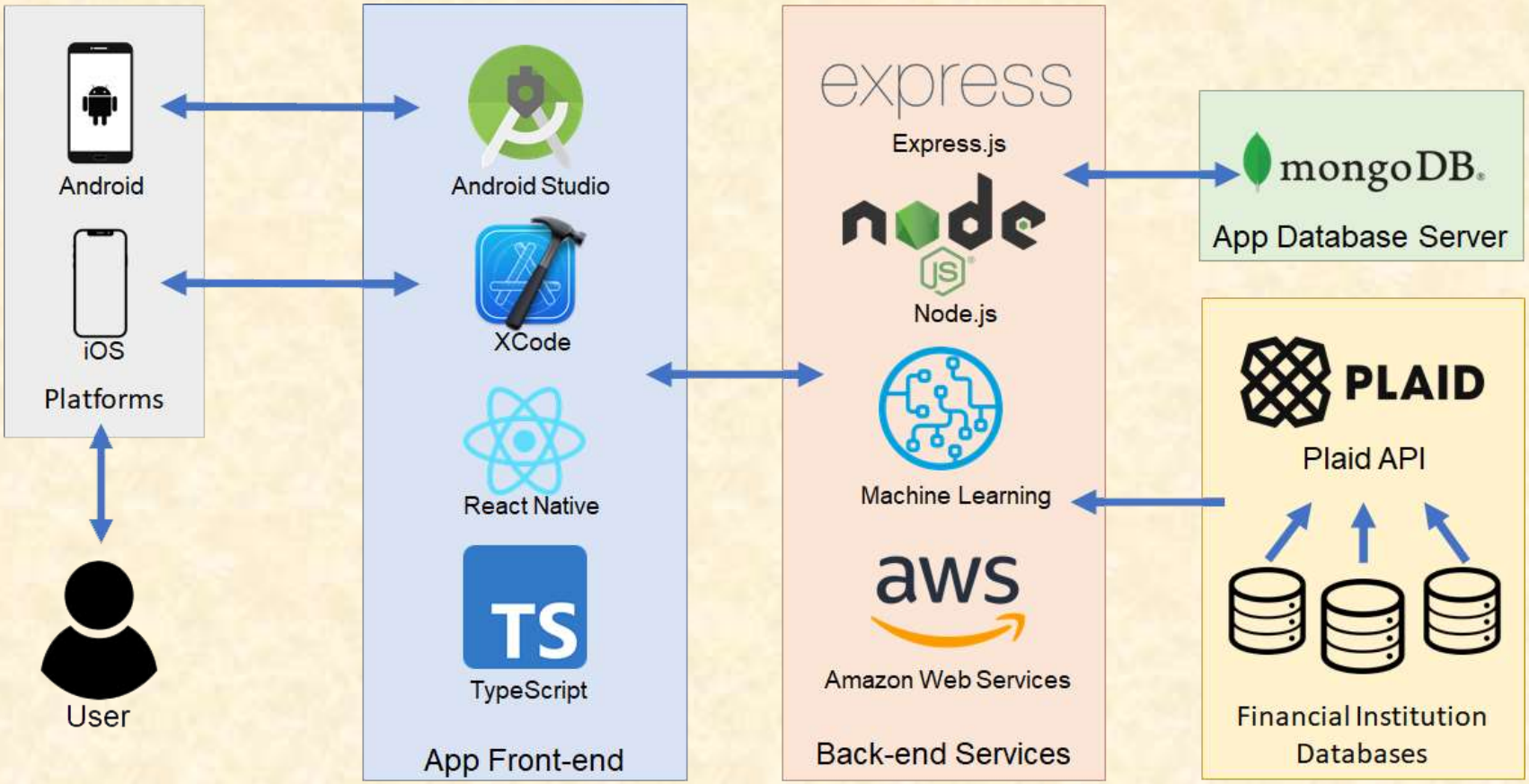
Figure 6: Detailed Charity Information

Figure 7: Horizontal Display Mode

Technical Specifications

- Using Machine Learning and AI, build a Recommendation Engine to find charities that match a user's interests
  - Utilize content based and collaborative filtering methodologies to provide the most accurate results
  - Translate user data from MongoDB and Plaid into a usable format for use in filtering algorithms
  - Finding patterns between user data and other users
- Create recommendation homespace for ML generated charities
  - Like and delete buttons will factor into the ML algorithm and be used to further tailor the curation of charities

System Architecture



System Components

- Software Platforms / Technologies
  - ***React Native*** - being utilized to serve both iOS and Android applications through the same codebase.
  - ***Visual Studio*** - we will be utilizing this IDE for React development.
  - ***Plaid API*** - used to connect Spave app to financial institutions.
  - ***Node.js*** - a JavaScript runtime environment that will be used for server-side development.
  - ***Express.js*** - a back end web application framework for Node.js.
  - ***MongoDB*** - a document-oriented database program used to host and manage the different charities within the Spave app.
  - ***Android Studio*** - the development environment for the Android OS .
  - ***Xcode*** - the development environment for macOS.
  - ***Typescript*** - the main language used on the front-end of the Spave app.
  - ***AWS*** - a cloud computing platform that will be used to host the Spave app.

Risks

- Acquiring Suitable Feedback from Real Users
  - Description: Our sample of users may be limited to specific demographics due to families and friends being the primary source of application feedback/testing.
  - Mitigation: In the case of low confidence in feedback, we will work with client to get the application in front of a more diverse group of users.
- Utilizing Plaid Transactional Data & Spending Analysis Tool Data
  - Description: Integrating the already existing Spending Analysis Tool into the new functionality, while simultaneously analyzing raw transactional data obtained through the Plaid API.
  - Mitigation: Effectively utilize data from both these sources by identifying specifically what each data source can contribute to the machine learning processes, such as revealing frequency of purchases at certain establishments or percentage of donations.
- Integrating Different Machine Learning Techniques Conjointly
  - Description: A combination of collaborative filtering and content-based filtering will need to be properly employed to make accurate recommendations.
  - Mitigation: Research filtering methods of existing applications to see if these methods could apply to the functionality desired for the project.

Questions?