MICHIGAN STATE UNIVERSITY

Project Plan

Predictive Rich Cards - Gemini

The Capstone Experience

Team GM

Phillip Prescher
Andrew Davenport
Michael Suszanne
George Wang
Tanay Salpekar

Department of Computer Science and Engineering
Michigan State University

Fall 2016

From Students…
…to Professionals

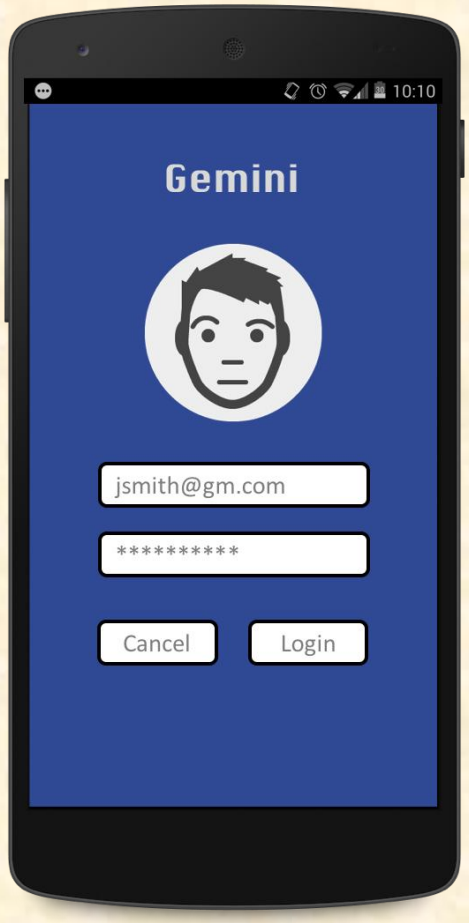
Functional Specifications

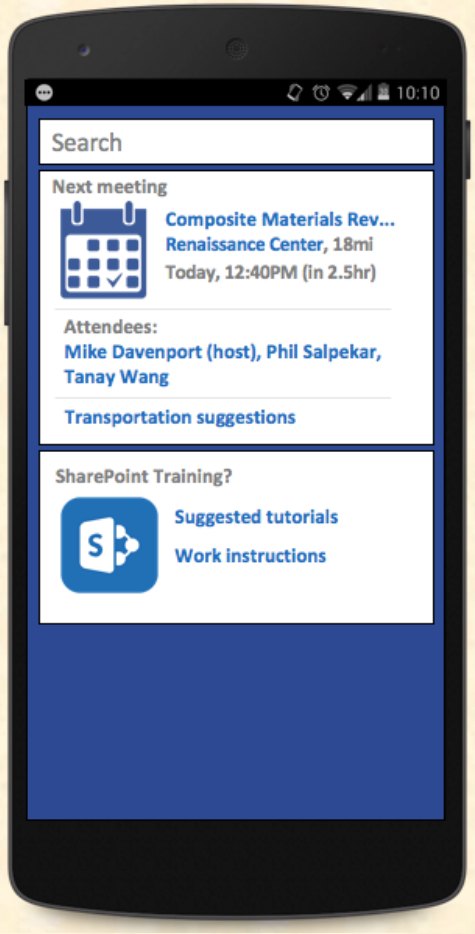
- Mobile Application for GM employees
- Uses predictive learning to help with employees' daily lives
- Learn user tendencies to deliver "cards" of information
- Ex: If an employee typically uses the shuttle as mode of transportation, automatically build shuttle into their schedule when they have an upcoming meeting

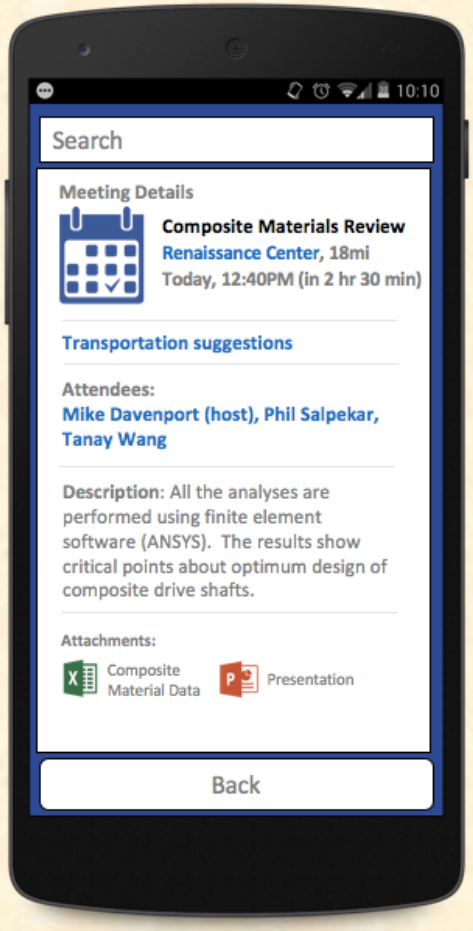
Design Specifications

- Mobile application features “cards” that show snippets of upcoming information that is relevant to the employee
- Push notifications for urgent information
- Example: For a meeting, a card is composed of meeting time, attendees and their profiles’, documents, and most importantly the transportation method

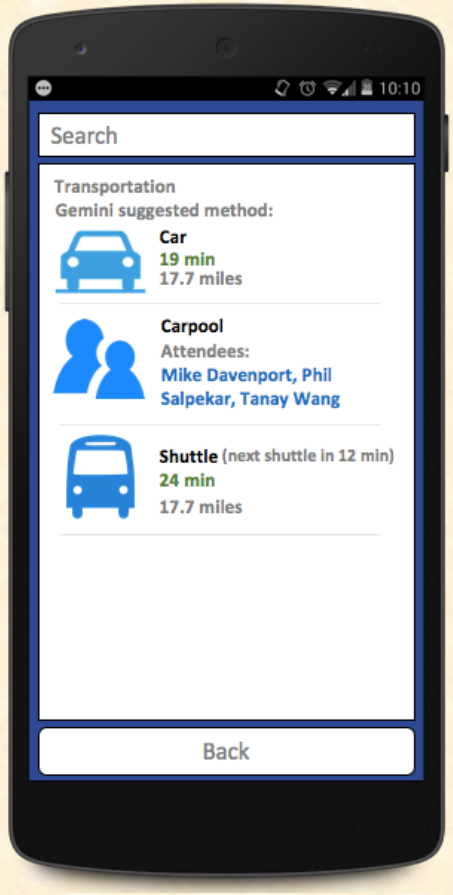
Screen Mockup: Gemini

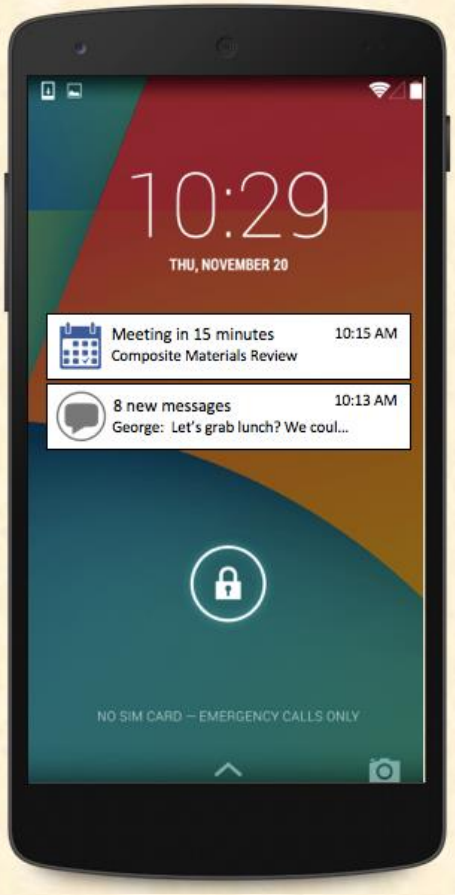






Screen Mockup: Gemini 2

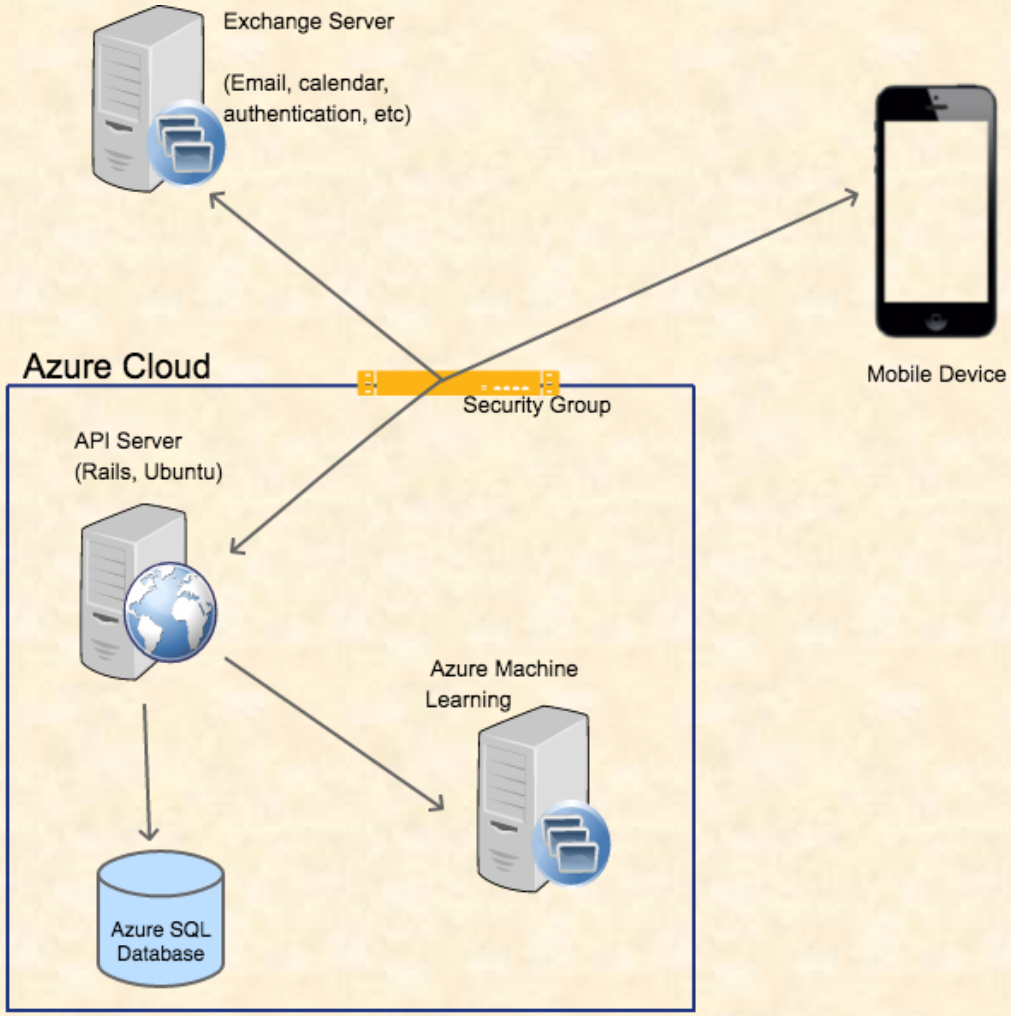




Technical Specifications

- Azure Cloud
  - Rails API – acts as communication medium between mobile application and data
  - PostgreSQL Database – persistent data store
  - Machine Learning – continuous processing of data to learn habits written in Python
- Xamarin
  - Cross-platform development mostly written in C#
- Exchange Server
  - Serves sample data that replicates GM internal environment

System Architecture



System Components

- Hardware Platforms
  - Azure Cloud
    - Two virtual machines – API and database
    - One Machine Learning environment – Azure-specific server for machine learning
  - Capstone Rack Server
    - Exchange Server that our application uses to get the employee's data
  - Android/iOS
    - Devices to use and test our mobile application
- Software Platforms / Technologies
  - Xamarin
    - C# cross-platform mobile development for iOS and Android
  - Ruby on Rails
    - API server to communicate between client application and all data
  - Python & Machine Learning
    - Unique Azure machine learning software and interface utilizing Python

Testing

- Unit testing for API
  - Login, fetching data, correct predictive analysis using rspec in Ruby
- Manual System Tests
  - Manually create meeting and ensure push notifications and cards appear on the attendee's device
- Performance Tests
  - Ensure our design and infrastructure provide timely delivery of cards and notifications

Risks

- Accurate Sample Data
  - Replicating an accurate model of GM data they use internally
  - For each new test case or data model, get approval from a technical GM contact
- Valuable Machine Learning
  - Need a large amount of data that a correctly configured machine learning environment can process. Can anything be predicted from this data?
  - Start creating and testing large amounts of sample data as early as possible
- Up-to-date Client Application
  - Machine learning algorithms might be too slow to predict urgent user items. Mobile application needs to present information only when it is relevant
  - Mitigate by continuous testing and ensuring all machine learning is done well before the project end date.