MICHIGAN STATE UNIVERSITY

Project Plan

Customer App Review Dashboard

The Capstone Experience

Team Ford

Paul Friesen
Noah Keppers
Drew Morgan
Jordan Schroeder
Mingkai Yang

Department of Computer Science and Engineering
Michigan State University

Fall 2018

From Students…
…to Professionals

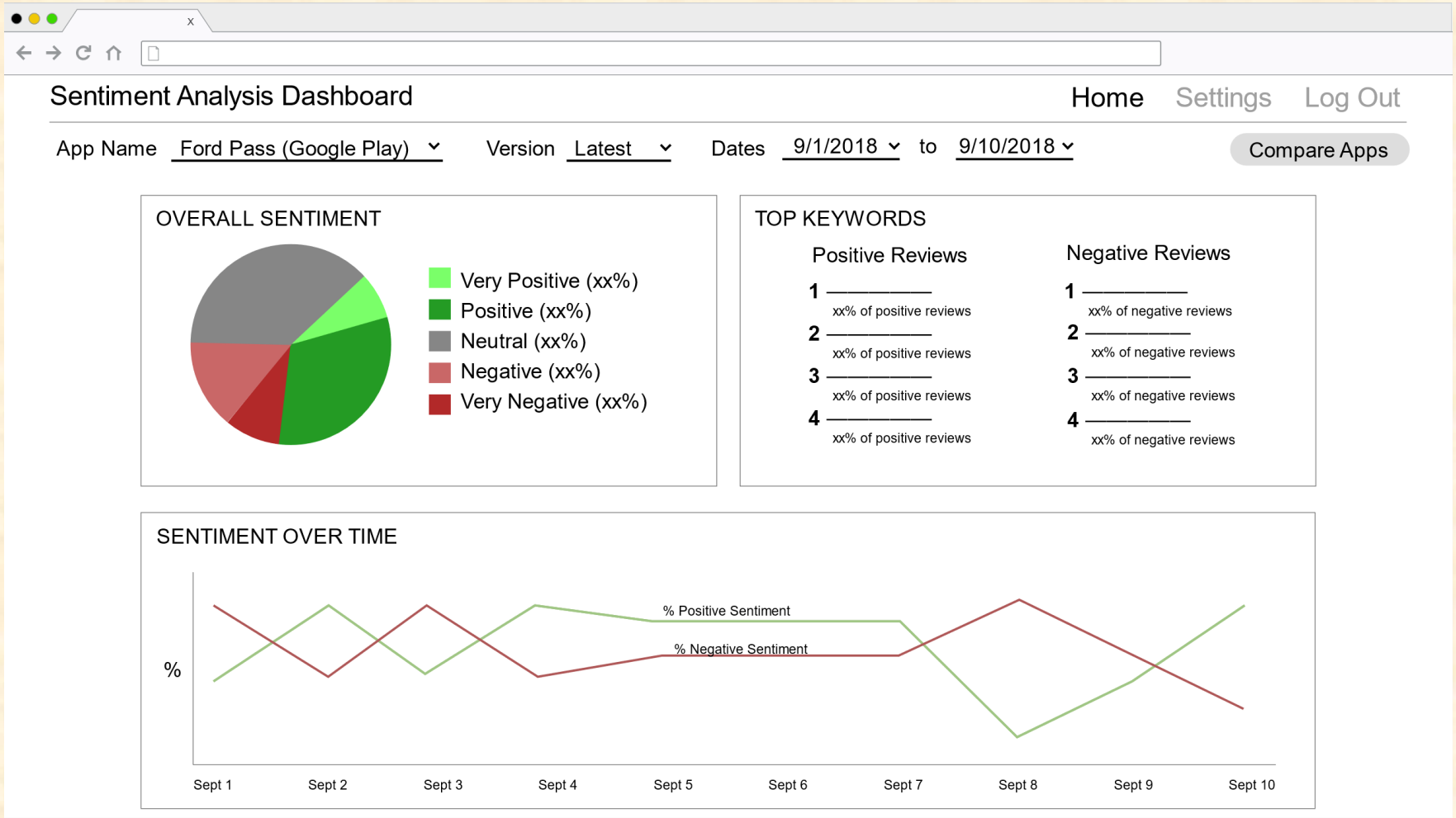
Functional Specifications

- Aggregate and analyze app review sentiment
- Present sentiment data in multiple formats
  - Scheduled Slack Bot reports
  - Web interface with a variety of views
- Compare multiple apps and versions
- Identify repeatedly mentioned app features

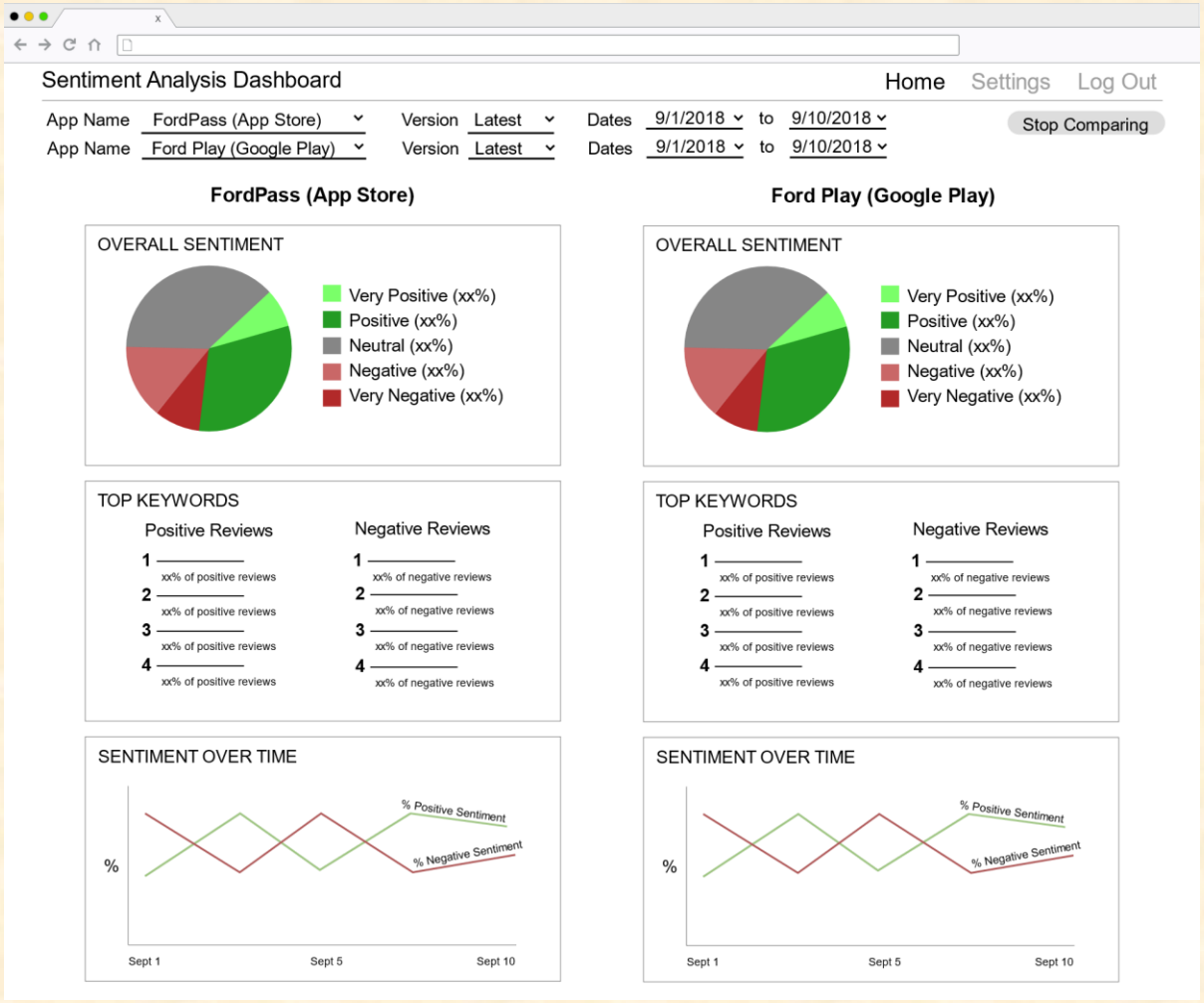
Design Specifications

- Web dashboard
  - Configurable interface allows for viewing statistics of a single app or comparing multiple apps/versions
  - Includes administrative portal for configuring both web interface and Slack Bot
- Slack Bot
  - Automatically outputs app sentiment report on a dedicated channel
  - May be explicitly summoned

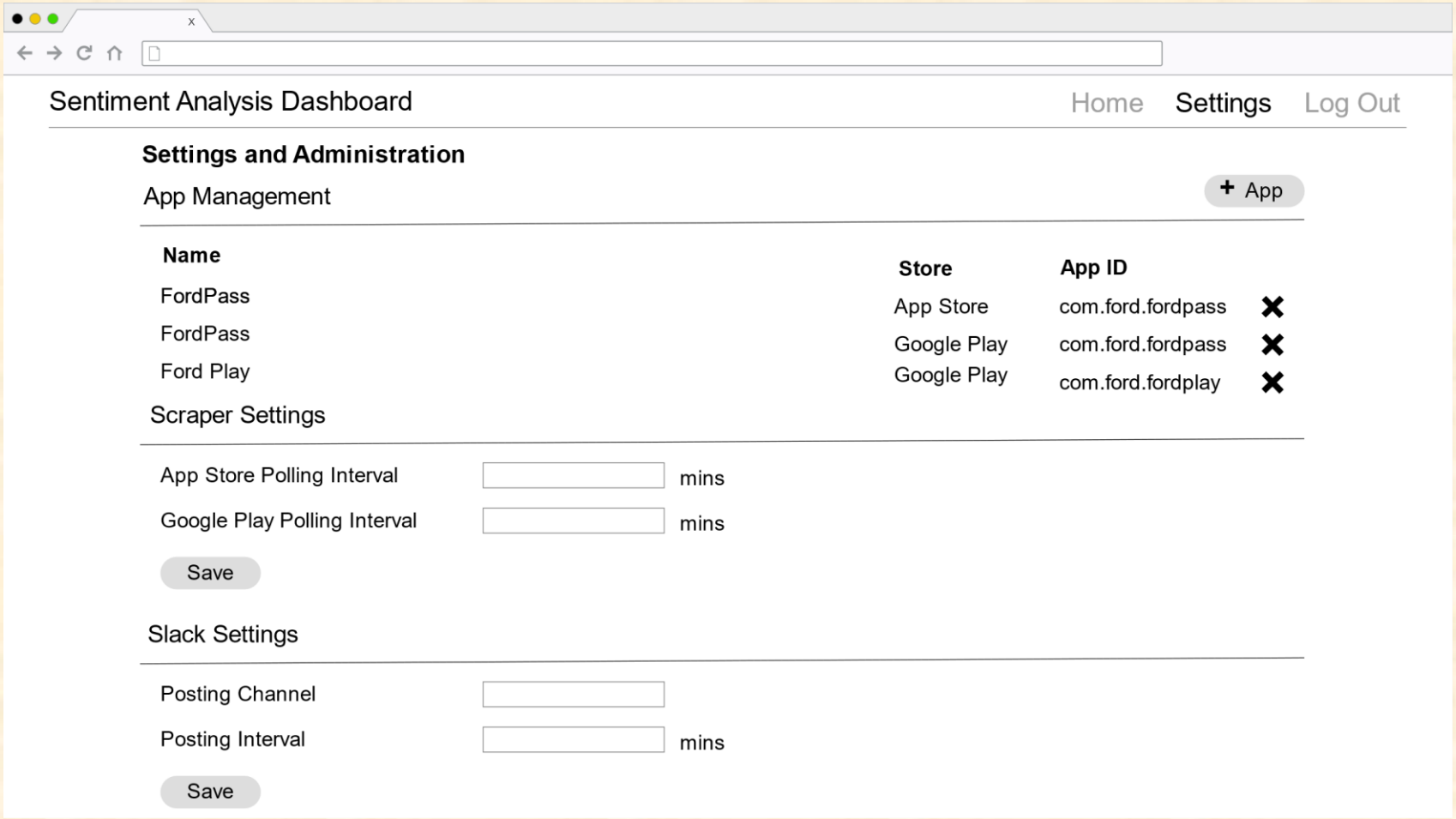
Screen Mockup: Single App Dashboard



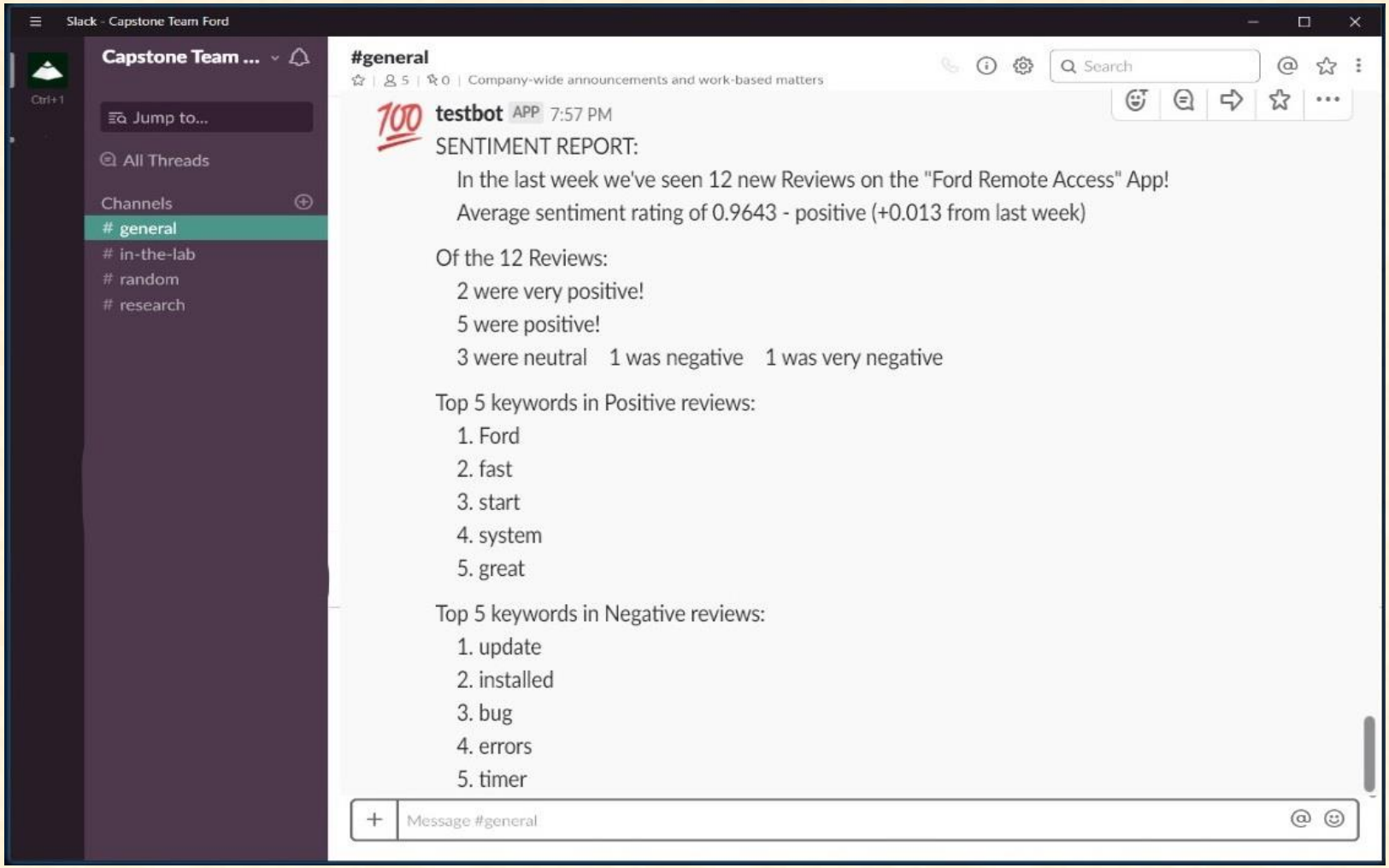
Screen Mockup: App Comparison Dashboard



Screen Mockup: Settings Portal



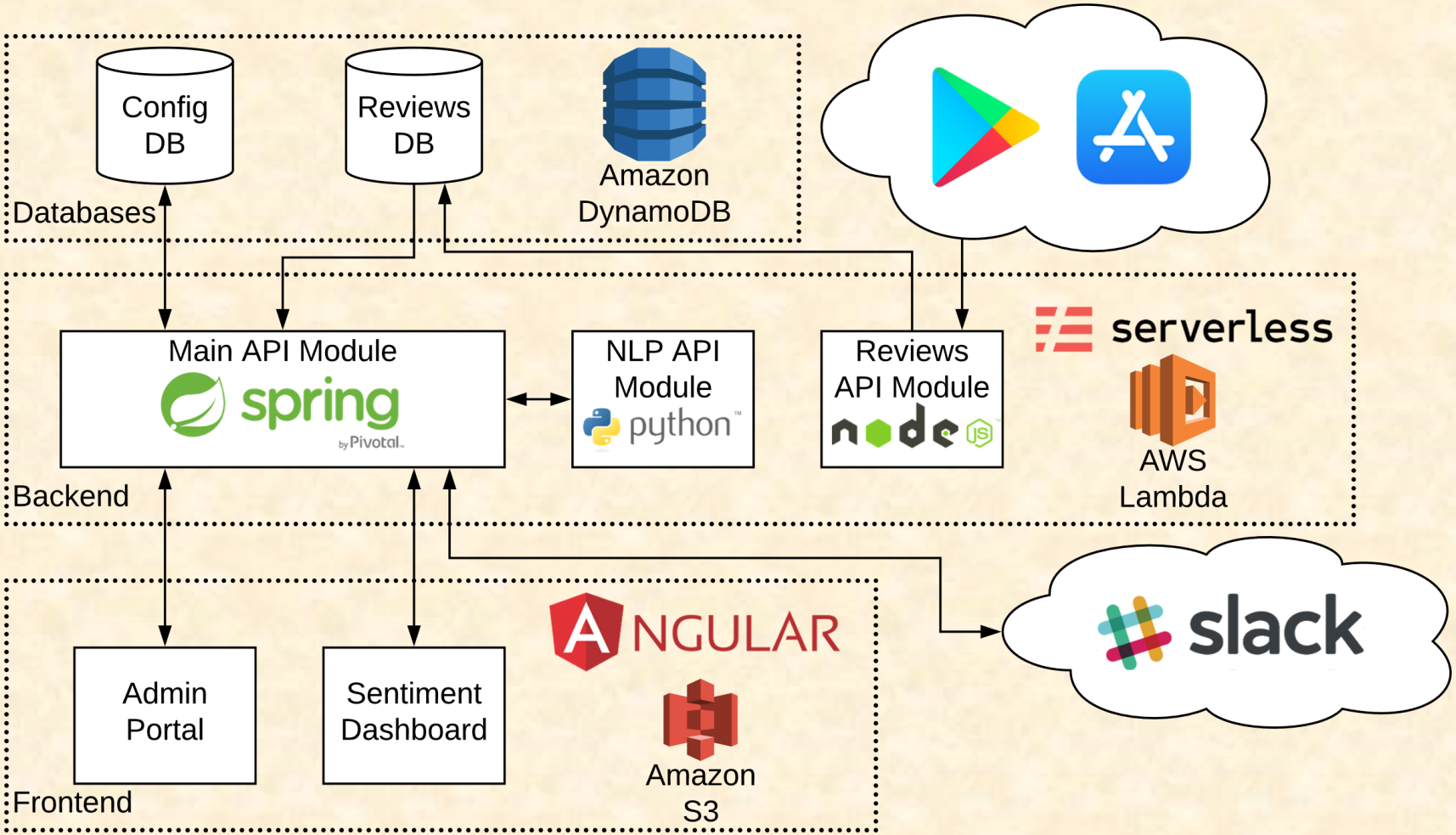
Screen Mockup: Slack Bot Output



Technical Specifications

- Backend
  - Java/Spring API
  - Node.js app store scraping module
  - Python NLTK sentiment analysis module
- Frontend
  - Angular web interface
  - Slack API bot

System Architecture



System Components

- Hardware Platforms
  - AWS Lambda
  - Amazon DynamoDB
  - Amazon S3

System Components

- Software Platforms / Technologies
  - Spring Cloud Function
  - Python Natural Language Toolkit
  - Node.js
  - Angular
  - Slack API
  - Serverless Framework
  - Travis CI

Risks

- Risk 1
  - Description: Inconsistent review data between stores and scraping tools
  - Mitigation: Develop a UI and DB schema that can handle varied information or implement functionality to scraper(s)
- Risk 2
  - Description: Web dashboard user authentication
  - Mitigation: Use a flexible authentication platform that integrates with LDAP (currently looking into Amazon Cognito)
- Risk 3
  - Description: Integrating multiple languages into one backend API
  - Mitigation: Use API Gateway and Lambda to allow isolated modules to communicate
- Risk 4
  - Description: Spring performance with Lambda
  - Mitigation: Increase allocated Lambda resources and test AWS Labs's suggested performance fixes

Questions?