MIDS W205

Course Project
Updated: 9/20/15

Instructors:

Jari Koister, jari@ischool.berkeley.edu

Dan McClary, dan.mcclary@ischool.berkely.edu

Karthik Ramasamy, karthik@ischool.berkeley.edu

Arash Nourian , nourian@ischool.berkeley.edu

Manos Papagelis , papaggel@ischool.berkeley.edu

Introduction

This project is part of the final exam of this course. The intent of this course is that students will demonstrate that they meet the objectives of the course. The project involves defining a problem, understanding the storage and processing needs (short term and long term) of a solution and selecting an appropriate technical approach. Finally the students are expected to create an initial proof-of-concept implementation of the solution that demonstrates their understanding of stitching together a viable end-to-end solution.

The students are asked define their own problem, find the appropriate data, and define the solution. The spirit of the solution should be that of industrial-strength architecture that enables development of a prototype, but also a providesa path to a production strength system. If students so request, instructors may provide suggestions for a problem statement.

Guidelines

- You need to identify a business or research problem based on an existing or new data set. There are no constraints on the data as long as all privacy or confidentiality constraints are met.
- You need to implement a process that computes the result in a repeatable fashion. Hence, it cannot just be a one-time computation. It must be sufficiently easy to kick-off a new end to end execution of the process.
- The result of the processing should be accessible for review through some kind of serving layer and presented in a form that would make sense in an intended real world scenario.
- You can pick any of the technical solutions discussed in the course as long as you can justify why you picked that solution. The justification must be grounded in a real world use case.

Evaluation and Acceptance Criteria

Deliverables and criteria

1. **A proposal presentation** in PPT, Keynote or Google Slide. You should think about the proposal as something that you would be presenting to an executive for a go/no-go decision, determining the funding of the project. You are expected to justify solving the problem and motivating the solution you are proposing.
2. **A final presentation**. A presentation of the problem, the final product, and a roadmap for improving the solution with increased usage and increasing data size.
3. All **Code** Submitted to Github per submission guidelines.
4. The instructor should be able to clone**, build and run the project**.
5. All **required documents and presentations in Github**. The repo should be completely self-contained and creating in accordance with submission instructions.

6. All **know limitations** with respect to scale etc. should be documented in the **final presentation**.
7. There should be a **runnable instance** of the solution.
8. Analyze the **complexity and storage** needs for the application; include this in the **final presentation**.

The following table provides guidance on how projects are graded and what aspects that are being considered in the review.

Aspects	**Description**	**Grading**
Base Requirements	Buy in	As described in deliverables and criteria section
Scope	The purpose of this aspect is to level the projects against each other in terms of scope and difficulty. A very difficult project should have more weight on the final grade than a very simple project, give other aspects are equal. Multiplier: multiply the sum of the points with the factor.	**Core Requirement:** 1. Meaningful use case involving an end-to-end data flow using at least one data set. Easy, few or no external dependencies. 2. Practical useful scope with potential user value, at least one real world data source. *1.0 factor.* **Scope that warrants additional recognition:** 3. Advanced analytics using complex computation. 4. Real-time analytics (*Velocity*) 5. Multiple real data sources that are merged or cleaned or both. *(Variety)* 6. *High volumes of data handles by solution (Volume)*
Functional (weight 40%)	This aspect is to determine if there actually is a functional working implementation that can be ran. *Max 100 pt*	**Core Requirement (50 pt):** 1. Functional code that can be executed and demonstrated. 2. System implements a base functionality specified in project proposal. **Extra Requirements (50 pt):** 1. Involves more than one data set that are combined. 2. Involves regular updates of data. Either in batch or streaming. 3. Involved dirty data that is being cleaned as part of the processing. 4. Has a clear business oriented purpose and problem in mind.
Design/Architecture (40%)	This aspect is to determine if the design and architecture selected adheres to principles learnt in the class. *Max 100 pt*	**Core Requirement (50 pt):** 1. A design that meets functional needs and is built on a set of technologies that represents a reasonable design for the intended solution. 2. Follow architectural principles and technology choices covered in course. 3. Express trade-offs and rational for decision. **Extra Requirements (50 pt):** 1. Implementation that scales according to some defined dimensions. 2. Implementation has clear processing and service layers. Implementation has a clearly architected data ingest layer.
Final Presentation (20%)	This aspect is to determine how well presented the project is. Is it clear that students understand what they build and why. *Max 50 pt*	**Core Requirement (25 pt):** 1. Basic documentation of architecture and functionality. 2. Presentation as defined by project instructions. **Extra Requirements (25 pt):** 1. Description of how to scale the solution. 2. A description of how to evolve the project, how in corporate additional data, future processing and technology needs.

Problem

1. You should formulate the specific problem and use case for the system/application.
2. It does not need to be a big data problem, but it should involve complexity along some dimensions such as the size of the data (Volume), the quality and variety of the data (Variety), the speed at which data arrives and need to be analyzed (Velocity).
3. It is permitted, but not required, to select a problem that requires advanced processing such as machine learning algorithms.
4. You can make assumptions about the data, number if processes, users etc. One such assumption could for example be that the data will be cleaned to a certain degree. But all such assumptions must be explicitly defined, and when appropriate reflected in the solution as tests, schemas etc.

Suggestions

Start by finding a data set. Based on the data set identify an interesting insight from the data using exploratory analysis. Implement a processing pipeline that can process and derive the insight repetitively. Determine what how frequently the result should be computed and how frequently you expect the data to be updated. Based on this determine what kind of architecture you need. Make sure that the architecture you choose can scale. If there are limitations on scale document them and check with an instructor that the limitation is acceptable.

Have clear deadlines so that you do not get stuck in a specific phase of the project.

Implement a steel thread of the solution quickly (a steel thread is a subset of the functionality implemented end-to-end)

Milestones

Week 4: Form groups & select problem area. Prepare slides (2-5 slides) for 10 minute presentation of your goals, challenges, how you will acquire your data. Also present what information organization challenges do you face as well as your initial plan to complete the project.

Week 6: Present a proposal. A proposal (1-2-pages) must be sent to the instructor with sufficient detail of the problem being addressed and the supporting research that data can be acquired and organized.

Week 11: Progress Report (1-2 -pages): description of first component of the project idea summary and justification, a partial description of data acquisition and organizing strategy and justification, tools/third party libraries description usages and initial performance evaluation on the adopted data acquisition strategy.

Week 15: Project Presentation: Presentation of your project in class and final submission. Allow instructor to run your solution. Submit any code and final report which includes (See above for details on acceptance criteria):

1. Overview of the problem being addressed.
2. Acquisition and organization of information for analytics.
3. The overall architecture of the solution and necessary implementation details.
4. The results of the project.