Setup Requirements:

- Sun JDK 1.6.0
- Maven 3 (http://maven.apache.org/)
- IDE of your choice (NetBeans is recommended)

Problem description:

Product owners need a way to measure various user behaviors in order to make informed decisions. To collect the data, you're tasked to write a REST based web service that mobile clients (Android and iPhone) can post data to.

Requirements:

- Use Protostuff (http://code.google.com/p/protostuff/) to define the protocol so that object models can be generated on each platform.
- Data is collected by submitting POST requests to /v*X*/events (where *X* is the protocol version number) with the metrics event data encoded as protobuf or JSON as specified by the Content-Type header.
- Data is retrieved by submitting GET requests to /v*X*/events/id where *id* is the id of the metrics event. The event is formatted based on the Accept header and JSON by default if not specified.

For POST requests, the server returns the following responses:

2XX	Data successfully stored in the database.
4XX	Content-Type is application/x-protobuf or application/json but the post data fails to deserialize.
4XX	Content-Type is not application/x-protobuf or application/json.
5XX	Failed to store submitted data in database.

For GET requests, the server returns the following error responses:

2XX	The metrics event exists and is returned in the response body.
4XX	The id specified in the path is not stored in the database or the path is invalid.
4XX	The Accept header is present but doesn't include application/x-protobuf, application/json or the wildcard */*.
5XX	Failed to retrieve submitted data from database.

Part 1:

- Implement InMemoryDatabaseAccess to use for testing.
- Implement POST and GET handlers in ClientMetricsResourceV1.
- The POST handler(s) should allow both protobuf and JSON encoding as input depending on the Content-Type header.
- The GET handler(s) should allow both protobuf and JSON as output depending on the Accept header and if no Accept header is present, or set to "*/*", output should be JSON by default.
- The GET request handlers should only accept numerical id's and return a 404 otherwise.
- Modify metrics.proto so that clients can submit arbitrary key/value pairs as part of the ClientEventData object.
- Add test cases to AbstractClientMetricsResourceTest demonstrating successful storage and retrieval plus error cases.

Part 2:

Jersey has a serialization framework that allows you to write readers and writers that convert byte streams into objects before passing them to the resources. The servlet handlers accept the objects directly instead of InputStreams.

- Implement ProtostuffMessageBodyReader and ProtostuffMessageBodyWriter.
    - You'll have to use reflection to get the schema.
    - Use CacheBuilder to create a LoadingCache that caches the schemas to avoid using reflection each time.
- In ClientMetricsResourceV2, implement handlePost and handleGet.
- Use the same tests written for ClientMetricsResourceV1 to test ClientMetricsResourceV2.

Evaluation:

You will be evaluated on:

- Design choices (ex. choice of error responses)
- Completeness of test cases
- Conciseness of code
- Documentation