ECE 313 (Section B)
Mini Project #1
Final Due Date: Friday, September 6, 11:59 PM

Software as a Service (SaaS) is an emerging paradigm in delivery of software services. In SaaS, an independent software vendor hosts its software in a data center, which the vendor may own and manage, or may acquire from a third party. Enterprises interact with applications delivered as SaaS using Internet browsers and other standard communication means (e.g., ftp and email clients), eliminating the need for upfront investment in software products, infrastructures, and expensive maintenance costs.

An example that illustrates the growing market for SaaS is that of CRM (Customer Relationship Management) and CPG (Custom Package Good). Such systems process very large amount of business-oriented data, such as stock inventory, product availability, and sales on a daily basis. In 2012, 40% of CRM Systems sold were cloud/SaaS-based, CPG are the second biggest spenders on CRM with a year growth of 12.5% over 2013.

Despite the interest of industry in SaaS products, little is known about the failures that such platforms can experience. Erratic behavior can in fact become a barrier to the future development of SaaS and jeopardize both customers and the service providers that rely on such platforms for their businesses.

Your task consists in analyzing data on operational failures of a state-of-the-art SaaS-enabled CPG platform. You are an analyst working at the company ACME and your duties consist in quantifying how much a specific system deployed by ACME is reliable, i.e., if it fails, how frequent it fails and how it fails. For doing that, you use data extracted from computer logs collected at runtime by the system under analysis. Considered computer logs contain information whether a given operation performed by the considered system succeeded or failed at time T.

The considered system processes a large volume of data received from 42 different customers scattered across 24 countries around the world. The processed data consist of files containing daily, weekly and monthly sales, inventory, or order details. On average, 3000 files/92 GB are received and processed each day. Operations consist of massive batch transformation (e.g., cleansing and harmonization) and processing of raw business data in uploaded files. Processing results are fed to several downstream business intelligence applications. The platform adopts state-of-the-art virtualization technology in a dedicated data center and runs over several virtual machines executing Microsoft Windows Server 2008, .NET platform, and SQL Server over VM-ware products.

The dataset that you will analyze in this project includes failures that occur during the processing of customers' data. They can have a number of causes, including software bugs and unexpected values in customers' data. Computer logs collected over 3 months of in-field operation are have been used to extract the data you will use in this project.

Description of the System

The platform is deployed over several processing nodes (application servers) and database servers. Functionally, it contains 7 key processing components or services. Each of the 7 key stages has sub-stages, which total to 18 processing stages. Some of the stages can have multi-thread functionality enabled within them, and overall up to 6 parallel threads can be run in the application. Several processing phases are carried out through a specific workflow that depends on the type of data to process (e.g., weekly sales or order details)

- Data validation (P1): Data are periodically received from the clients. This stage checks the structure and validity of the data format (e.g., number and type of fields).

- Data notification (P2): This component sends notifications to various stakeholders when errors are encountered, for instance, concerning the data format.

- Data verification (P3): Received data are scanned for potential viruses and verified at the semantic level for correctness and consistency according to pre-specified formats.

- Data transformation (P4): Business transformations are applied to harmonize the data.

- Data commit (P5): Data are sorted and moved to their respective final repositories, where these are committed.

- Data archiving (P6): All data are archived for audit purposes.

- Data distribution (O4): The processed and harmonized data are sent to downstream systems for inventory management and business intelligence.

Each stage of the platform is implemented as a set of .NET applications. All services are independent of one another and perform their functions by periodically polling for relevant files, status code changes, and DB entries. The system integrates several Commercial Off The Shelf (COTS) components for common functionalities, like data manipulation and transformation.

Log Data

The impact of an error can range from a minor problem to a platform hang/crash or database corruption. Two types of failures are present in the considered data. The first, referred to as *user data failures*, are due to errors caused by user mistakes (e.g., a wrong/corrupted client file) and are detected during the first stage (P1) of the

computation (data validation stages). Files failing the validation stage are not processed by the platform. Instead, they are sent back to the client with a detailed report on the causes of the rejection. The second, referred to as *platform failures*, can manifest in any stage of the computation, and are caused by system-related issues, including residual bugs, operating system errors, and hardware/network problems.

In the three months of collected data, the system received 275,790 files from 42 customers in 11 different countries.
Logs consist of several files stored as plain text generated by specific logging procedures implemented as part of the platform code. Entries are collected as client data traverses through the processing stages and include detailed information such as the size of the file, the number of records in the file, client information, time, and, in the case of a failure, the computing stages involved in the failure and logged exit code.

The data is structured in the following fields:

Submission Time	Computing Stage in the Failure	Computation Start Time	Computation End Time	Failure Cause	Failure Details	Failure Type
7/1/12 0:02	IT1	7/1/12 0:02	7/1/12 0:02	File Not Received	Went over cut-off time	USER DATA FAILURE
7/1/12 0:58	IT4_L2	7/1/12 0:59	7/1/12 0:59	System Error	Package Validation/Execution Failed.	PLATFORM FAILURE
7/1/12 0:59	IT4_L2	7/1/12 0:59	7/1/12 0:59	System Error	Package Validation/Execution Failed.	PLATFORM FAILURE
7/1/12 1:01	IT4_L2	7/1/12 1:01	7/1/12 1:02	System Error	Package Validation/Execution Failed.	PLATFORM FAILURE
7/1/12 1:01	IT4_L2	7/1/12 1:01	7/1/12 1:01	System Error	Package Validation/Execution Failed.	PLATFORM FAILURE