From: Manager McWoman
To: Matthias^2
Subject: Spreadsheet Spec

The language you should choose for this project is Golang 1.11, so as to stay in sync with the other developers working on the company's other projects. For this spreadsheet project, we want five primary data definitions: Spreadsheets, Equations, Cells, Functions, and (pre-defined) Numbers/Primitives. The data setup should be as follows: a **Spreadsheet** (struct) contains a 2D array of Cells (struct) based on a coordinate system where columns are assigned (A, B, …, Y, Z, AA, AB…) and rows are numbered 1 to N, where N is the number of rows in the Array. A **Equation** (interface) is one of the following: a **Cell** (struct), a **Function** (struct), and **primitive constants**. A Cell contains a single Equation. A Function contains two Equations and one operator (for now, just one of **addition** or **multiplication**). Note that we are working with a recursive data definition. Numbers are just language primitives. Note that a Cell does not contain any information about its own relative or absolute position in the spreadsheet. Given these data definitions, the spreadsheet will be the source of most of the functions that users will expect: since functions can have recursive definitions across cells (ex. "A1 + B1"), the Spreadsheet must have a method to evaluate Cell values, which themselves may be recursive (ensure no circular cell definitions). A Spreadsheet should be resizable, and should preserve relevant top-left-most data during resizing. A Spreadsheet should be creatable from an existing 2D array of Cells, which should be as easy as assigning the underlining 2D array to be what is given (and assigning that array's size to your max row/col count). The Spreadsheet will be what changes Cell values, and in doing so may evaluate other Cells based on their coordinate system, creating a recursive evaluation method that evaluates child Cells when referenced (as would be if A1 contained the equation "1 + C3 + B1").