

# CS60 Patient Safety App: A Volume-based Enteral Feeding Calculator

Enteral Feeding App Team  
Alison Jones & Parker Okonek  
CS461 - Fall 2019

## Abstract

Current enteral feeding calculations are prone to human error. This document outlines the requirements for an enteral feeding rate calculator with a focus on accuracy, ease of use, and simple, open source design. This document explains the current enteral feeding calculation method and its issues, and offers the benefits of the application's approach, its features, and what those features will accomplish.

## I. Introduction

### A. Purpose

The volume-based enteral feeding calculator replaces the paper tables currently used when calculating feed rates for both gastric and small bowel enteral feeding. The calculator reduces error by requiring the source numbers to be entered and automatically produced, unlike the current method which requires the result to be produced by hand. The goal of this conversion is to reduce both error and time taken in getting the results.

### B. Scope

Paper tables have been designed for nurses to use to calculate the catch up rate. The table is a commonly used rate-based system with a max rate cap based on the method of feeding. Many hospitals don't even use this method and just had calculate the catchup rate. The current method leaves some room for error and could be improved/automated.

The application and its source code will be released on GitHub. Both the compiled binary and the source code will be appropriately licensed for its own open source distribution and to be license compatible with any libraries, code, or other intellectual property used to create the application.

The platform for the project will be determined from surveying nurses to determine whether mobile or desktop is more convenient and widespread for projects of similar scale and niche.

## II. Specific Requirements

### A. External Interfaces

The enteral feeding calculator will interface with local storage on the machine to store any generated error logs and to save values which are not likely to change after being first set. The calculator will also display to the user's screen and input devices via a UI library to interface with the operating system's screen drawing routines and to accept pointer and text input from the user.

### B. User Requirements

The user interface will be minimal and easy to read. The application will have no configuration window or views other than its main interface, and on its main interface will have the text inputs for the different input values: time of feeding stop, time of feeding restart, reset time of feeding, and daily feeding volume.

Time inputs will be only selectable options in drop down menus. Any time with a PM will have 12 added to its value automatically before being used for calculations. A time with PM at the end cannot have an hour value greater than 12. Volume inputs will accept only integer values. A dropdown menu will be used to display all available feeding types: Nasogastric, Nasojejunal, Percutaneous Endoscopic Gastronomy, Jejunostomy. The selected feeding type will be displayed on the closed dropdown and on the informational banner in the top left. At the bottom of the UI will be the output value and a calculation button that updates the output value, if valid.

### C. Functions

The program will take the user's input for the various fields of the finished algorithm. The application will use these values to compute the enteral feeding rate for the remaining time of the day but not exceed safe maximum rates. The program will have modes for both gastric and small bowel feeding, which will alter the maximum rates and the calculations for rate. The app should be able to get date and time information for these calculations. As the user enters and updates information, it should be updated where applicable. The variable representing the time of feeding reset should remain constant between the application closing and being opened, but remain configurable when opened.

### D. Software Quality

The enteral feeding calculator will need to have support for multiple versions of its target platform. Medical settings often have a variety of versions for device operating systems. The application will support legacy and current versions of Windows, from Windows 7 to Windows 10.

The software should not crash, freeze, or return erroneous results over normal use. The results returned by the program should never exceed the maximum safe values for their kind of enteral feeding, even with erroneous input. Results should also be consistent over up time of the application and regardless of the uptime of the operating system.

The user interface should indicate when a value is accepted or rejected, and offer further clarity through its interface with minimal clutter.

### III. Verification

The project should decrease the time to calculate an enteral feeding rate by at least 25% with a 50% decrease being unlikely. The application should increase accuracy of nurse calculations, though the current metrics of nurse errors are not known. After being proven to work, the final project will hopefully make it into clinical testing. This is the true test to verify how effective the project is. The application should ultimately be well received and have a lowered chance for error. We can test this by comparing the percent error before and after implementation.

### IV. Gantt Chart

Fig. 1: Gantt Chart

#### Items

1. Go over the rate table/ process that nurses go through when calculating rate based feeding or catch up rate for feeding.
2. Research which interface to use. Contact nurses and learn whether a phone or computer application would be better. Also ask about inputs/ first design check at this time.
3. Research App development based on chosen interface.
4. Develop Algorithm/ math and logic side to app.
5. Integrate algorithm into app framework.
6. Add time based input for application.
7. Ensure user input is saved locally on device to preserve data.
8. Work on app display and usability (pretty stuff)
9. Check app design again with nurses/ projected users.
10. Update app design based on feedback. (repeat as necessary).
11. Finalize Application and publish? Research what is needed to do with this.
12. Check whether we can start implementation and research.
13. Start is we can.
14. Final Report and Wrap-up

### V. Changes

#### A. User Requirements

Previous Text	Current Text
Time inputs will allow only numerical values, a colon, and optionally AM/PM inputs. Any time with a PM will have 12 added to its value automatically before being used for calculations. A time with PM at the end cannot have an hour value greater than 12. Volume inputs will accept only numerical values.	Time inputs will be only selectable options in drop down menus. Any time with a PM will have 12 added to its value automatically before being used for calculations. A time with PM at the end cannot have an hour value greater than 12. Volume inputs will accept only integer values.
A checkbox will be used to indicate whether the feeding type is gastric or small bowel, with a large banner indicating the current state.	A dropdown menu will be used to display all available feeding types: Nasogastric, Nasojejunal, Percutaneous Endoscopic Gastronomy, Jejunostomy. The selected feeding type will be displayed on the closed dropdown and on the informational banner in the top left.

#### B. Software Quality

Previous Text	Current Text
Medical settings often have a variety of versions for device operating systems. If the mobile platform is chosen, the software will need to support multiple versions of both the Android and iOS mobile operating systems. If the desktop platform is chosen, the software will need to support legacy and current versions of Windows, from Windows 7 to Windows 10.	The application will support legacy and current versions of Windows, from Windows 7 to Windows 10.