act ProposeTimeLine

ActivityInitial

[fullInit==true]

Full Initialization

Priorities are set according the length and the expected end dates.
Note that priorities (weights) will change during calculation, depending on how MSE (mean squared error) has changed.

Iteration Initialization

Reset variables needed for calculation
(mostly those start with calc...)

Prepare the Time Unit List (timeline for each worker).
A Time Unit represents a week.
And it is filled up during calculation how much hours a worker should work on a task (in this week)

ActivityFinal

If calculation of one iteration takes too long there is propably an logical error in definitions.
(which will be displayed to user)
Possible Reason Example:
Task1 is predecessor of Task1

Choose Task

Choose a task randomly considering task weights

Set Task Chosen for Worker and TimeUnit

[relationOk]

[!relationOk]

If chosen task can't be done by the worker
(this may have several reasons
e.g. Max%ofTask already reached)
then choose another one

This loop is necessary, cause in some cases worker may work on different (more than one) task in a week

Staff Member Loop

[atLeastOneMaNecessary && everyTaskFinished]

[x<steps]

Time Unit Loop

Check if Plan of this iteration is finished

One Iteration:
Has unique task priorizations and the result is an error

Calculate the Error (MSE)

Recalculate Prioritiy Weights

[error <= bestError]

ActivityFinal

ActivityFinal

Exit with new Best Error