Following are some of the briefs of the solution approach adopted. The solution divided in five different steps.

Please note, in order to achieve reproducibility, the file download also done via programming. First few line of the run_analysis .R carries out download and sub-sequent unzipping of the data dump provided. Please refer line [26-32] of run_analysis.r. This program accepts directory name, where user want to download the data dump and subsequent processing to happen. The final tidy dataset will be available under **'dir/ UCI HAR Dataset/'** , where dir is the user provided directory. Name of the file containing tidy dataset is **'final.txt'.**In users doesn't provide did parameter , everything will happen in current working directory. In case users doesn't provide parameter , everything will happen in current working directory.

Step 1:

1.a)

Features.txt read into a data frame and subsequently transposed to form the header of the final data frame. Here two additional column "subject","activity" also appended to the transposed data frame, which eventually forms the final header, to be used later.

```
----------------------------------------------------------------------------------------------------------
## Reading Files
  ### 1. features.txt
  features<-read.table("features.txt")
  x<-as.vector(features[,2])
  ## Transposing the feature name to column, so that it can fit the final dataset as header
  ## Also adding two extra column for subject and activity
  header<-t(x)
  h<-append(paste("V",c(1:561),sep="),c("subject","activity"))
  header<-cbind(header,c("subject"))
  header<-cbind(header,c("activity"))
  colnames(header)<-h
----------------------------------------------------------------------------------------------------------
```

1.b)

This step has two different sub steps, as mentioned below.

1.b.1:
Test dataset, which contains 30% of the data read here. Three different data.frame created, one each for 'x_test.txt', 'subject_test.txt' and 'y_test.txt'. After this all of them merged (column binding) into single data frame using cbind().

```
----------------------------------------------------------------------------------------------------------
  ## Reading test files - X_test.txt
  setwd(testdir)
```

```
 xtest<-read.table("X_test.txt")
 ## Reading subject files - subject_test.txt - to extract subject
 subject<-read.table("subject_test.txt")
 colnames(subject)<-c("subject")
 ## Reading activity files - y_test.txt - to extract subject's activity for test sample
 y<-read.table("y_test.txt")
 colnames(y)<-c("activity")
 ## Adding subject and activity to test data.frame
 xtest<-cbind(xtest,data.frame(subject))
 xtest<-cbind(xtest,data.frame(y))
```

1.b.2:
Here we deal with Train dataset.Similarly as above three data.frame also created here, each for ‘x_train.txt,’ ‘subject_train.txt’ and ‘y_train.txt’. Once done, all of them merged (column binding) using cbind()

```
 ## Reading train files
 setwd("..")
 setwd(traindir)
 xtrain<-read.table("X_train.txt")
 ## Reading subject files - subject_train.txt - to extract subject
 subject<-read.table("subject_train.txt")
 colnames(subject)<-c("subject")
 ## Reading activity files - y_train.txt - to extract subject's activity for train sample
 y<-read.table("y_train.txt")
 colnames(y)<-c("activity")
 ## Adding subject and activity to train data.frame
 xtrain<-cbind(xtrain,data.frame(subject))
 xtrain<-cbind(xtrain,data.frame(y))
```

1.c)
Once 1.b.1 and 1.b.2 finished, outcome of both these steps merged together (row binding) to form ‘finaldf’ data frame. Additionally header created in step – 1 assigned to the column name of the data frame ‘finaldf’.

```
 ## Creating final dataset with test and train dataset created earlier
 finaldf<-rbind(xtest,xtrain)
 ## Adding the header created from features.txt and assigning it as rowname
 colnames(finaldf)<-as.vector(header)
 #dput(xtest,"final.txt")
 #dput(xtrain,"final.txt")
```

1.d)
Reads activity_label.txt and creates a data frame ‘activitylabel’. Subsequently ‘activitylabel’ and ‘finaldf’ gets merged to form ‘mergedata’. Basically “megeddata” contains an additional column (activity_lable) alongwith “finaldf” dataset. Here merging done based on key ‘activity’ (common in both the dataset)

Step 2:
These steps extract the entire column with word ‘mean ()’ or ‘Std ()’ suffixed to it. I am not including column, which contains word ‘mean’, or ‘std’ middle the column name text.

Step 3:
This step formats the entire activity labels.
e.g : after this step activity_label with value "WALKING_UPSTAIRS" will be shown as “Walking Upstairs” in the final dataset . A new function replace_all (defined at the top of code) to replace old activity _label with new one has been used.

Step 4:
Some minor adjustment to the column name of the dataset (outcome of step 3) has been made here.

```
-----------------------------------------------------------------------------------------------------
#### Extracting column name to format
  cn<-colnames(submission3)
  #### Replacing ^t with time , ^f with freq and () with ..
  cntime<-replace_all(cn,"^t","time")
  cnfreq<-replace_all(cntime,"^f","freq")
  cnfinal<-replace_all(cnfreq,"\\(\\)","")
  colnames(submission3)<-cnfinal
-----------------------------------------------------------------------------------------------------
```

Step 5:

Sqldf package used to calculate the avg () of each variable for each activity and each subject . And final outcome of step submission5 contains the tidy data, which eventually redirected to ‘final.txt’ file. ‘final.txt’ is the ultimate tidy dataset.