TimeMe

Software Requirements Specification

Ver. 1.0

May 23rd, 2012

Lucas, Kevin, Mario, Jesse

Prepared for
CS220 – Software Engineering
Instructor: B. Clarke
Spring 2012

Revision History

Date	Description	Author	Comments
March 7	Initial import	Lucas	Google Doc
March 14	Section 2 snippets	Everyone	Various
March 20	Section 2 content	Mario & Jesse	Section 2 paragraphs
March 21	Cleanup & Appendices	Kevin	ToC & Section 2
April 3	Section 1	Everyone	Various
May 1	Section 3.1, 3.2, 3.4, 3.5, 3.6	Everyone	Added and Edited Diagrams.
May 12	Section 3.3,3.4,4.2,4.3,4.1	Everyone	Added and Edited Diagrams and formatted document.
May 15	Cleanup	Everyone	All sections
May 16	Formatting	Jesse	Diagrams & ToC
May 19	All sections were edited.	Everyone	Various edits and diagrams were added to the whole document.
May 22	Appendices	Kevin	Added git info

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Date

Table of Contents

1. Introduction

TimeMe is full feature task tracking system. Users are able to enter task, organize tasks, record the time they spend on task, as well as view and print reports, related to their time. Within this document you will find the Hardware and Software requirements related to the system. You will also, find diagrams illustrating the proper use cases for user interaction, related system diagrams, and class references.

This document contains the business requirements of the application and references to all 3rd party modules. Additionally, you will find a description of the development environment and how to properly configure it for your development efforts.

1.1 Purpose

This document is written primarily for development purposes and for anyone wishing to know the intimate technical details of TimeMe.

1.2 Scope

TimeMe is full feature task tracking system. Users are able to enter task, organize tasks, record the time they spend on task, take notes related to their task, as well as view and print reports, related to their time.

TimeMe is intended to help users' better track where their time is spent. Furthermore, they Time will give users a better sense the time required for related task, so they may be able to prioritize effectively and plan their time wisely. The reporting system within TimeMe will give users an achievable history or their tasks for their own record keeping.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
JRE	Java Runtime Engine
UI	User Interface
OS	Operating System
SWT	Standard Widget Toolkit (Native Java GUI API)
JAR	Portable Java Archive File
TSV	Tab Separated Value File Format
GPL	GNU Public License

1.4 References

For all questions related to outside sources and documentation refer to Appendix (A.1 – A.4).

1.5 Overview

The rest of this document will contain the product description, Use Case Diagrams, System Diagrams, Class References and Diagrams. Also, contained in the rest of this document will be the non-Functional requirements for the system.

2. General Description

TimeMe is a task-tracking productivity tool designed which allows users to keep track multiple tasks while rapidly switching between them. It was created in response to the need for a way to keep exact time information and organize notes without excessive data entry. Once collected TimeMe maintains the data in a format which can be used to generate reports and saves the information in a format which facilitates portability. Because of these features TimeMe is a powerful task tracking tool which can allow users track billable hours, analyze productivity or revisit time spent during the day for any reason. It reduces time spent tracking work without sacrificing any detail in the data collected and is designed to run on multiple major hardware/software platforms and system architectures to further enhance portability of data.

2.1 Product Perspective

TimeMe is an open-source alternative to subscription-based task tracking software. It is dual-licensed under the Apache 2.0 License and the GPL v3 and available on a number of platforms and architectures.

2.2 Product Functions

TimeMe will have a list of active tasks, which the user will be able to add new tasks to or remove task from, once a task is completed. The user will be able to select an active task in order to view, in a task details panel, the current data related to the task. ie: Total time spent on the current task; The date the task was created; The full title of the task; Start timer; Stop timer; etc.

Once a task is selected and the user clicks Start timer, TimeMe will begin recording the elapsed time until the user clicks: Stop Timer. Afterward, TimeMe will store the elapsed time in a Tab Delimited configuration file.

When a task is first created, TimeMe will create a plain text “Notes” file for each task. This plain text file will be activated as soon as the user selects a task from the active task list, and it will editable at any time, from within the Notes tab. The purpose of the Notes file will be for the user to keep track of any special information related to the task or the billing of the task. This window may also be hidden to provide space on screen for the task at hand.

In the task management tab, users will be able to organize the priority of there tasks. There will be an up button to elevate the priority of a task and a down button to deemphasize the priority of a task. There will also be a list task, which can be review. Users will also be able to create new tasks from this tab as well as remove tasks. This portion of the window may also be hidden to provide space for the task at hand.

TimeMe will also provide a reports tab where documents can be created for billing and viewing purposes. These reports will be generated from within TimeMe and available to print.

2.3 User Characteristics

TimeMe’s target users are Support Specialist, Software Designer/Developers, Design Specialist, or any other professional who needs to keep track of billable time spent on projects or tasks. These users are not expected to be technically inclined, however they are expected to have

general office computing skills. Users will simply need to have enough knowledge to select a task and click the start timer button, and take notes should the need arise.

2.4 General Constraints

TimeMe must also be platform independent. Meaning, TimeMe cannot be written on any proprietary system Frameworks, i.e.: Microsoft .NET Framework. Therefore, TimeMe will be written in Java, utilizing the Sun Microsystems INC. Open-Source, cross platform, Java Runtime Engine (JRE). Also, all persistent data must be platform independent as well. So, rather than using Microsoft SQL, TimeMe's data must be stored in a MySQL Database, XML File, or tab delimited configuration file.

TimeMe will also need to be developed and test by the end of the Spring 2012 semester at Orange Coast College. The development team may not exceed 5 people and all documentation must be completed by the same deadline.

2.5 Assumptions and Dependencies

TimeMe's external dependency will be Oracle's (formerly Sun) Java Virtual Machine runtime powered by their Java Runtime Engine (JRE). For the purpose of this project all users will need to be run JRE version 1.6 or above to maintain compatibility with SWT. TimeMe will also be dependent on the JRE's Operating System compatibilities.

Oracle maintains JRE builds for x86 and x86_64 for Microsoft Windows, Apple OSX, and Linux. Other OSes such as BSD and Solaris as well as other platforms such as ARM, MIPS and PowerPC may be able to substitute OpenJDK. Compatibility with proprietary Java

implementations such as Microsoft Java and IBM Java will not be considered.

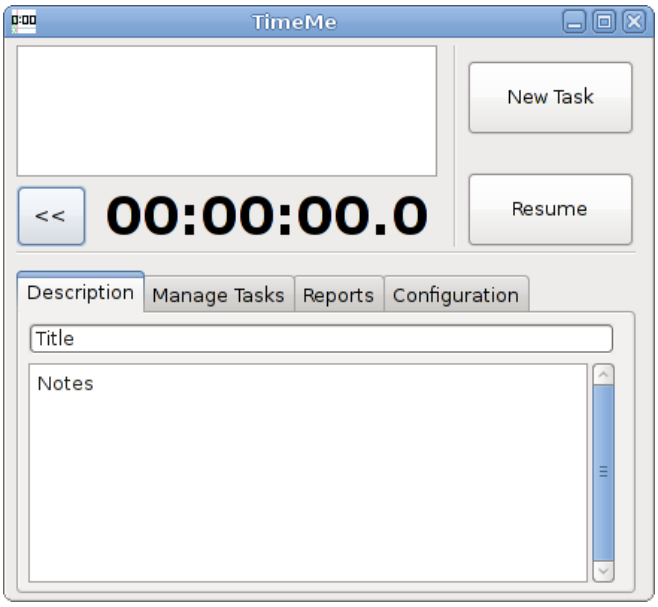
As of version 0.10, TimeMe is built on the SWT API from Eclipse Labs. This library is platform and architecture specific; however a number of the major platforms are supported. The advantage of SWT over SWING/AWT is the use of native widgets resulting in a more integrated and fluid experience. TimeMe will be packaged as a cross-platform JAR file supporting the Linux, OSX, and Windows platforms in both x86 and x86_64 varieties through the help of the jar-within-a-jar swtjar.jar helper and its accompanying ANT script.

If time constraints change, the TimeMe development team will have to continually evaluate workload and team responsibilities to manage the new requirements. The same holds true for changes to documentation requirements.

3. Specific Requirements

3.1 External Interface Requirements

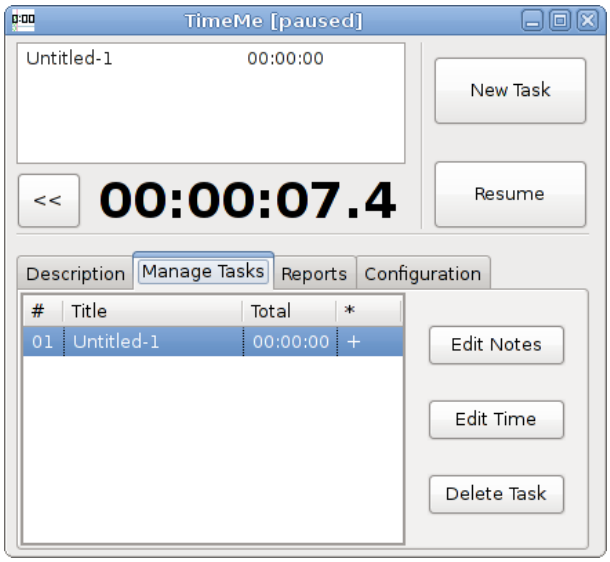
3.1.1 Full User Interface w/ Notes Tab Open



UI Figure 1

User Interface Object	Description
New Task Button	Clicking this button will create a new task for the user.
Pause/Resume Button	If the user is currently tracking a task, clicking this button pause the timer and records the time of the active task. Clicking the button again will restart the time for the active task. If no tasks have previously been started, this button will simply start the timer for the active task.
Resent Items Window	This window will show the user the 5 most recent tasks. The user will be able to select any task, switching the active tasks, and begin the timer (buy clicking on the Pause/Resume button).
Timer	The time will update according to the active task and once started will increment by the second.
Hide Button (<<)	This will show or hid the bottom half of the UI in order to maximize screen space for the user to work.
Notes Tab	The notes tab will be show the title of the currently active task. Also, it will have a text box which can be edited by the user.
Title Bar	The Notes title bar will display the title of the current task.
Notes Textbox	This textbox is provided so that the user may take special notes related to the currently active Task.

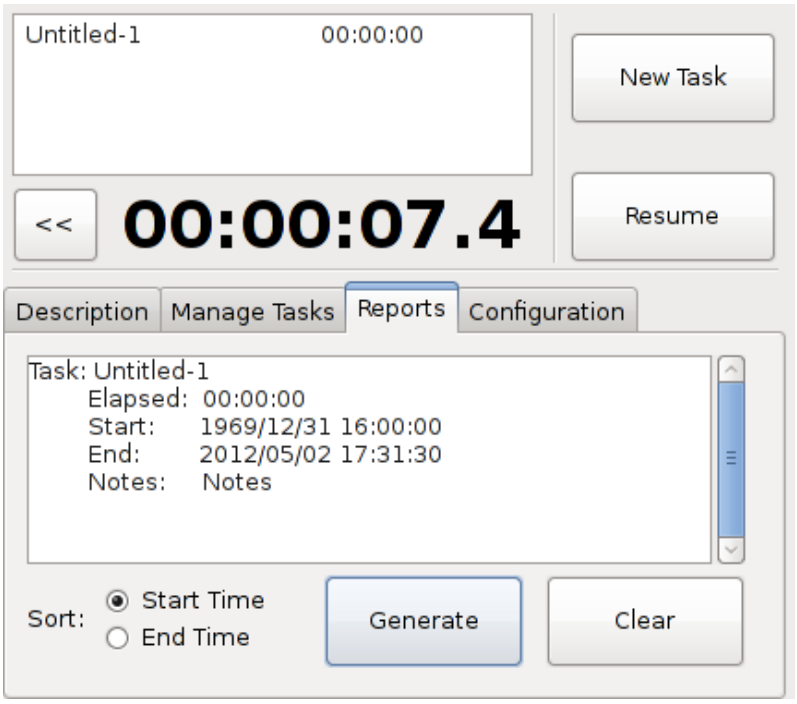
3.1.2 Full User Interface w/ Manage Tasks Tab Open



UI Figure 2

User Interface Object	Description
Manage Tasks Tab	This tab will give the users a list of all tasks which the user can track. The user will also be able to manage their tasks from this tab.
Task List	The task list view will contain all tasks which the user has entered into TimeMe. Highlighting a task will active the task. User will be able to switch back to the "Notes" tab in order to view notes for that task.
Edit Notes Button	This button will allow the user to edit the details of the task highlighted in the Task List, ie: Title.
Modify Time Button	This button will allow the user to modify the time for the task highlighted in the Task List.
Remove Task	This button will allow the user to remove (delete) the task highlighted in the Task List. Removing the Task will also remove all user data pertaining to that Task.

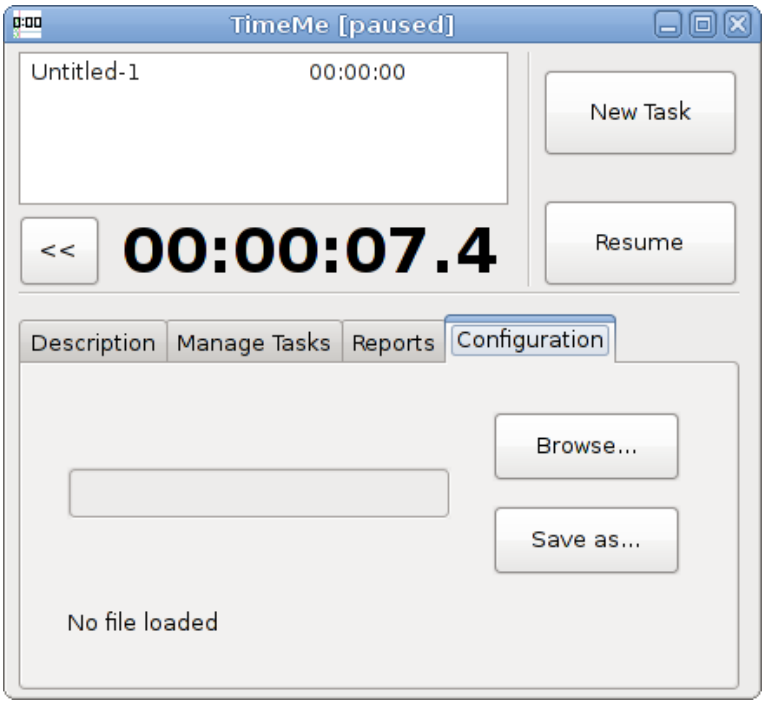
3.1.3 Full User Interface w/ Notes Tab Open



UI Figure 3

User Interface Object	Description
Data Files + Reports Tab	This tab will allow users to run a Task report.
Last Time Worked Option	This tells TimeMe to run a Task report in the order of the last time Tasks where worked on.
First Time Worked Option	This tell TimeMe to run a Task report in the order of the first time Tasks where worked on.
Run Report Button	Clicking this button will run a Task report based on the user option selected.

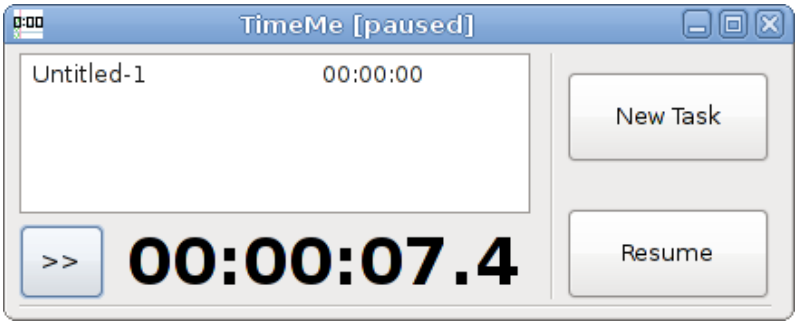
3.1.4 Full User Interface w/ Configuration Tab Open



UI Figure 4

User Interface Object	Description
Browse	This button will allow the use to browse for a configuration file on their local file system.
Save As…	This button allows the user to save their current configuration to local file system.
Textbox	Contains the string represent to the file path for the configuration file.
Status Label	Notifies the user if a file has been load properly.

3.1.5 Collapsed User Interface



UI Figure 5

User Interface Object	Description
Recent Task List	This list boxes contains a history of recent tasks
>> Button	This button collapse and expands the window.
New Task Button	This button creates a new task and expands the window.
Stop/Resume Button	This button stops the clock for the task and also resumes clock for the selected task.
Clock	Notifies the use about the amount of time spent on the current task.

3.2 Functional Requirements

3.2.1 Functional Requirement for New Task

Introduction

The New Task button stores the current task and creates a new one to work on.

Inputs

Inputs for the New Task button include the notes, title, task ID and time values of the previous task.

Processing

The New Task button checks to see if there is a current task and if one is present saves the task to the recent task table and the all tasks table. After previous tasks are recorded the a new empty task is created.

Outputs

Data from the previous task is entered into both recent task table and the all tasks table and displayed to the user. Generic text is entered into the title and notes fields.

Error Handling

If the title of the task has not been changed a it is changed to “untitled” and appended with a unique numerical ID.

3.2.2 Functional Requirement for Pause Task

Introduction

The Pause Button stops the running timer for the current task.

Inputs

None

Processing

None

Outputs

Set the state of the stopwatch to stopped.

Error Handling

This button is inaccessible unless the timer is running.

3.2.3 Functional Requirement for Resume Task

Introduction

The Resume Button Starts restarts the timer for the current task.

Inputs

None

Processing

None

Outputs

Set the state of the stopwatch to ticking.

Error Handling

This button is inaccessible unless the timer is running.

3.2.4 Functional Requirement for Edit Notes

Introduction

The Edit Notes button allows you to modify the notes of any previous task even if it is not in the current task list.

Inputs

The Edit Notes button receives the row ID of the task selected in the Manage Tasks tab.

Processing

Using the task ID from the Manage Tasks tab this button pulls all the information in that row and loads it as the current task.

Outputs

The current task is saved to the table and the data from the selected row is loaded into the current task object, the text fields and the stop watch. The current tab is switched to Description so that the notes are on screen immediately when the button is pushed.

Error Handling

None

3.2.5 Functional Requirement for Edit Time

Introduction

The Edit Time button allows you to modify the time of any previous task even if it is not in the current task list.

Inputs

The Edit Time button highlights the time field of the task selected in the Manage Tasks tab, prompting the user for input.

Processing

Calculate the millisecond equivalent of the user entered time.

Outputs

The displayed time field of the task selected in the Manage Tasks tab is set to the user input and the millisecond equivalent into the hidden elapsed time fields.

Error Handling

Values that are not time formatted are rejected.

3.2.5 Functional Requirement for Delete Task

Introduction

The Delete Task removes a task from the all tasks and recent tasks tables.

Inputs

The Delete Task button receives the row ID of the task selected in the Manage Tasks tab.

Processing

None

Outputs

Remove the task from the all tasks and recent tasks tables.

Error Handling

None

3.2.6 Functional Requirement for Start Time Task

Introduction

This radio button sets reports to be sorted according to their start times.

Inputs

None

Processing

None

Outputs

Set the report sort mode to Start Time

Error Handling

None

3.2.7 Functional Requirement for End Time Task

Introduction

This radio button sets reports to be sorted according to their end times.

Inputs

None

Processing

None

Outputs

Set the report sort mode to End Time.

Error Handling

None

3.2.8 Functional Requirement for Generate Task

Introduction

Generates a report based on task information.

Inputs

Every row is pulled from the all tasks list and is converted to a human readable task report including total elapsed time, start time, end time, title and notes for each task.

Processing

Tasks are converted to a human readable task report including total elapsed time, start time, end time, title and notes for each task. Task reporters are sorted based on the Start Time and End Time radio buttons.

Outputs

The report text is sent to the text field in the Reports tab.

Error Handling

None

3.2.9 Functional Requirement for Clear Task

Introduction

Clear the text field in the Reports tab.

Inputs

None

Processing

None

Outputs

An empty text string is entered into the text field in the Reports tab.

Error Handling

None

3.2.10 Functional Requirement for Browse Task

Introduction

Opens a browse dialog allowing the user to select and load saved task files.

Inputs

A file path is gathered from the user via the browse dialog.

Processing

The file at the selected location is read and parsed and loaded into the all tasks table

Outputs

All tasks read from the file will be loaded into the all tasks table.

Error Handling

Only .TSV files may be selected with the browse dialog.

3.2.11 Functional Requirement for Save As Task

Introduction

Save the current set of tasks to a file.

Inputs

Task information is read from the all tasks table and a filename is read from the user through the save dialog.

Processing

Rows are converted into strings for saving.

Outputs

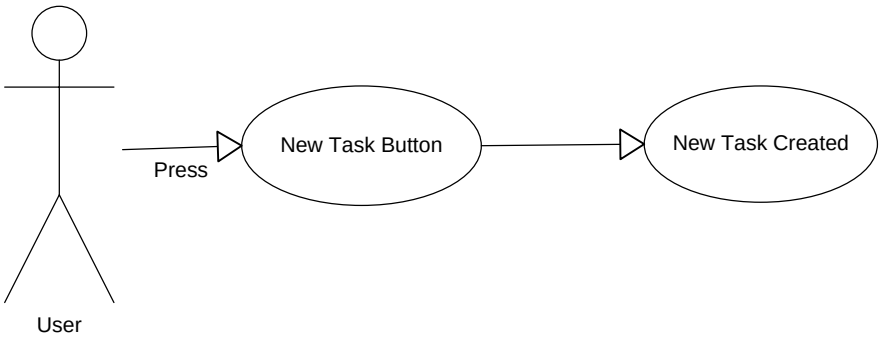
A file is written from the processed information.

Error Handling

None

3.3 Use Cases

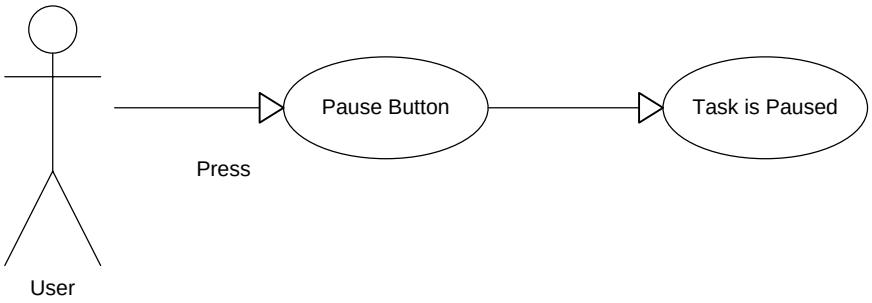
New Task Use Case



Use Case 1

New Task	Description
Actors	Field consultant, Task Table
Description	A consultant may pause a task that is currently running which will stop the task timer momentarily until the task is resumed.
Data	Task timer.
Stimulus	User presses Pause button.
Response	The current task has its timer stopped.
Comments	

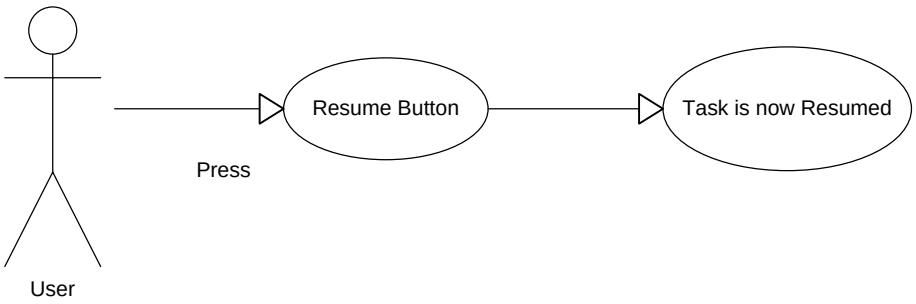
Pause Task Use Case



Use Case 2

Pause Task	Description
Actors	Field consultant, Task Table
Description	A consultant may pause a task that is currently running which will stop the task timer momentarily until the task is resumed.
Data	Task timer.
Stimulus	User presses Pause button.
Response	The current task has its timer stopped.
Comments	

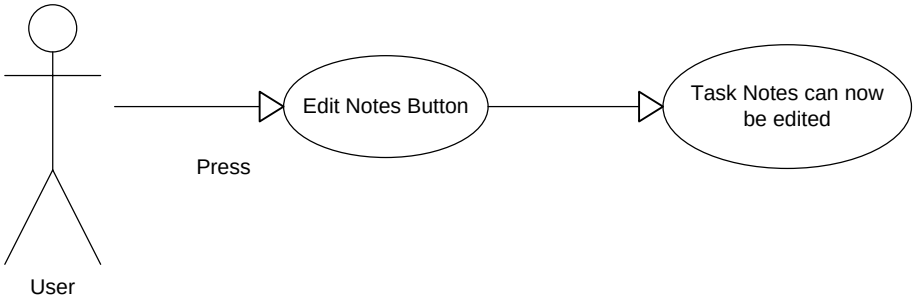
Resume Task Use Case



Use Case 3

Resume Task	Description
Actors	Field consultant, Task Table
Description	A consultant may Resume a task that is currently Paused which will start the task timer from the time it was paused.
Data	Task timer.
Stimulus	User presses Resume button.
Response	The current task has its timer resumed again.
Comments	

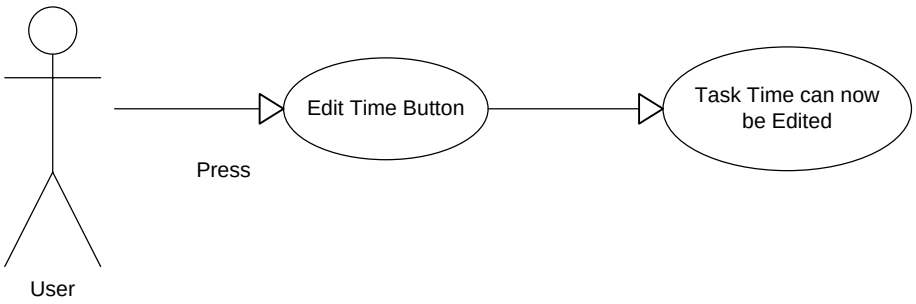
Edit Notes Use Case



Use Case 4

Edit Notes	Description
Actors	Field consultant, Task Table
Description	A consultant may edit the time of a task which will allow him to change the starting time, ending time, and the elapsed time of the specified task(s).
Data	Task Notes.
Stimulus	User clicks the Edit Notes button.
Response	The details tab opens and the Notes are now available for the consultant to add, remove or edit.
Comments	

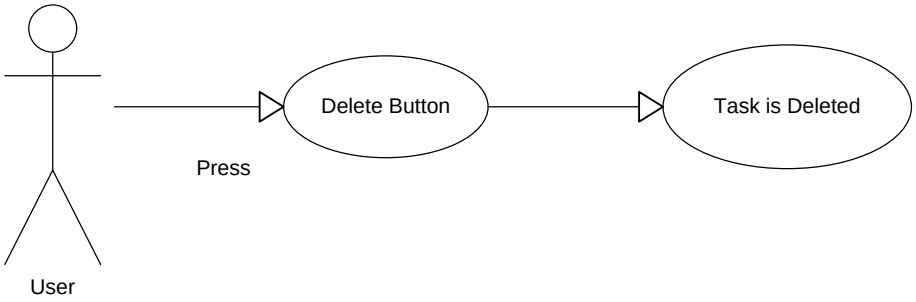
Edit Time Use Case



Use Case 5

Edit Time	Description
Actors	Field consultant, Task Table
Description	A consultant may edit the time of a task which will allow him to change the starting time, ending time, and the elapsed time of the specified task(s).
Data	Task time.
Stimulus	User clicks the Edit Time button.
Response	A dialogue box will appear that will allow you to change the starting time, ending time and time elapsed.
Comments	The user can choose which times to change.

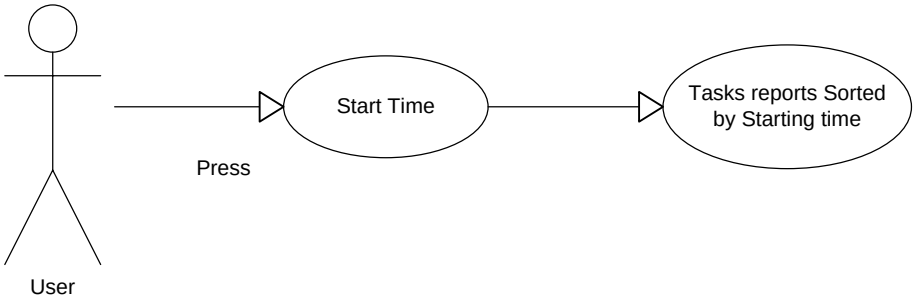
Delete Task Use Case



Use Case 6

Delete Task	Description
Actors	Field consultant, Task Table
Description	A consultant may delete a task(s) which will permanently delete the task wan all of its data from the program
Data	Task notes, Task times.
Stimulus	User clicks the Delete Task button
Response	It removes the row from the tabe, removes the task from the recent list (if applicable) and unselects the row
Comments	

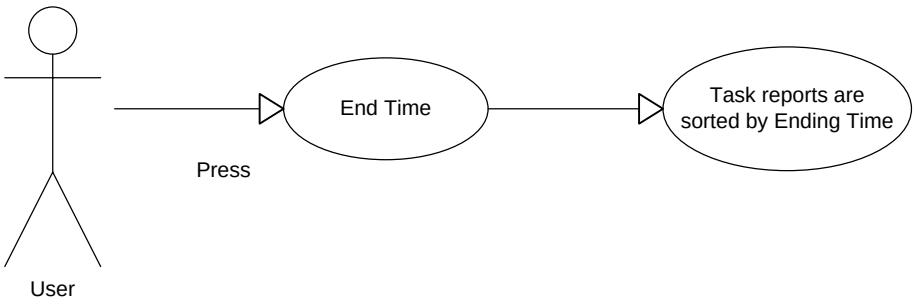
Start Time Task Use Case



Use Case 7

Start Time Task	Description
Actors	Field consultant, Reports Table
Description	A consultant may sort all the tasks in the order by which the tasks were created. Sorting the tasks by starting time would reorder the tasks in the Reports box from newest to oldest.
Data	Task starting time.
Stimulus	User clicks the Start Time button.
Response	The tasks are sorted in the order that they were created.
Comments	

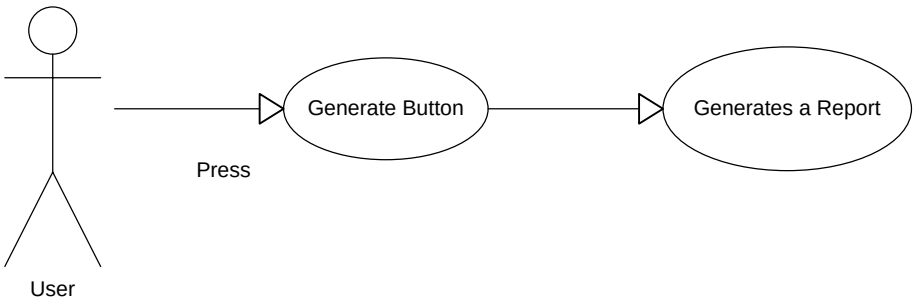
End Time Task Use Case



Use Case 8

End Time Task	Description
Actors	Field consultant, Reports Table
Description	A consultant may sort the tasks by the order that the tasks were paused.
Data	Tasks ending time.
Stimulus	User clicks the End Time button.
Response	Tasks are sorted by the order that they were paused.
Comments	

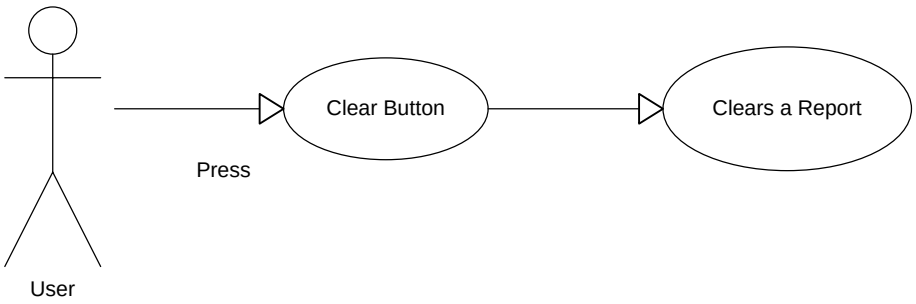
Generate Task Use Case



Use Case 9

Generate Task	Description
Actors	Field consultant, Reports Table.
Description	A field consultant can generate a report summary of any task(s) which will include start and ending times, notes, and titles.
Data	Generate exportable report
Stimulus	User clicks the Generate button.
Response	It will generate a report summary of the entire task session.
Comments	It will either be ordered by start time or end time.

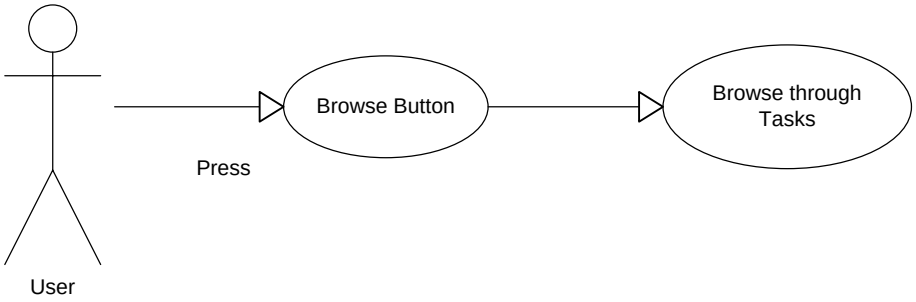
Clear Task Use Case



Use Case 10

Clear Task	Description
Actors	Field consultant, Reports Table.
Description	A field consultant can clear the report of a task which would leave the Report box empty and makes it ready for another report to be generated.
Data	None
Stimulus	User clicks the Clear button.
Response	Clears text box
Comments	Read only, doesn't change anything.

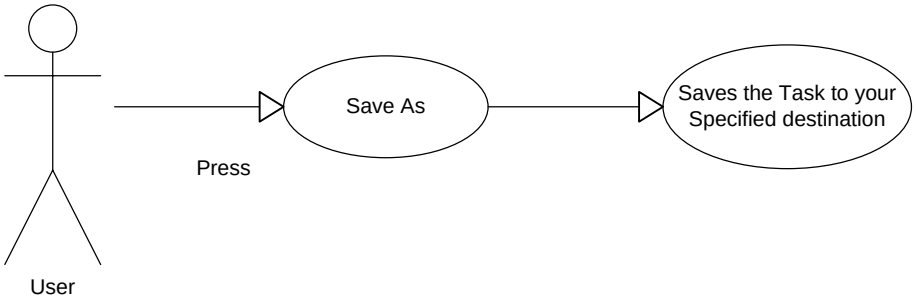
Browse Task Use Case



Use Case 11

Browse Task	Description
Actors	Field consultant, Configuration table.
Description	A field consultant may browse their files to load tasks that have been previously saved.
Data	Saved files.
Stimulus	User clicks the Browse button.
Response	Opens up document dialogue box which allows you to browse and select you previously saved task files.
Comments	Only opens .tsv files.

Save As Task Use Case

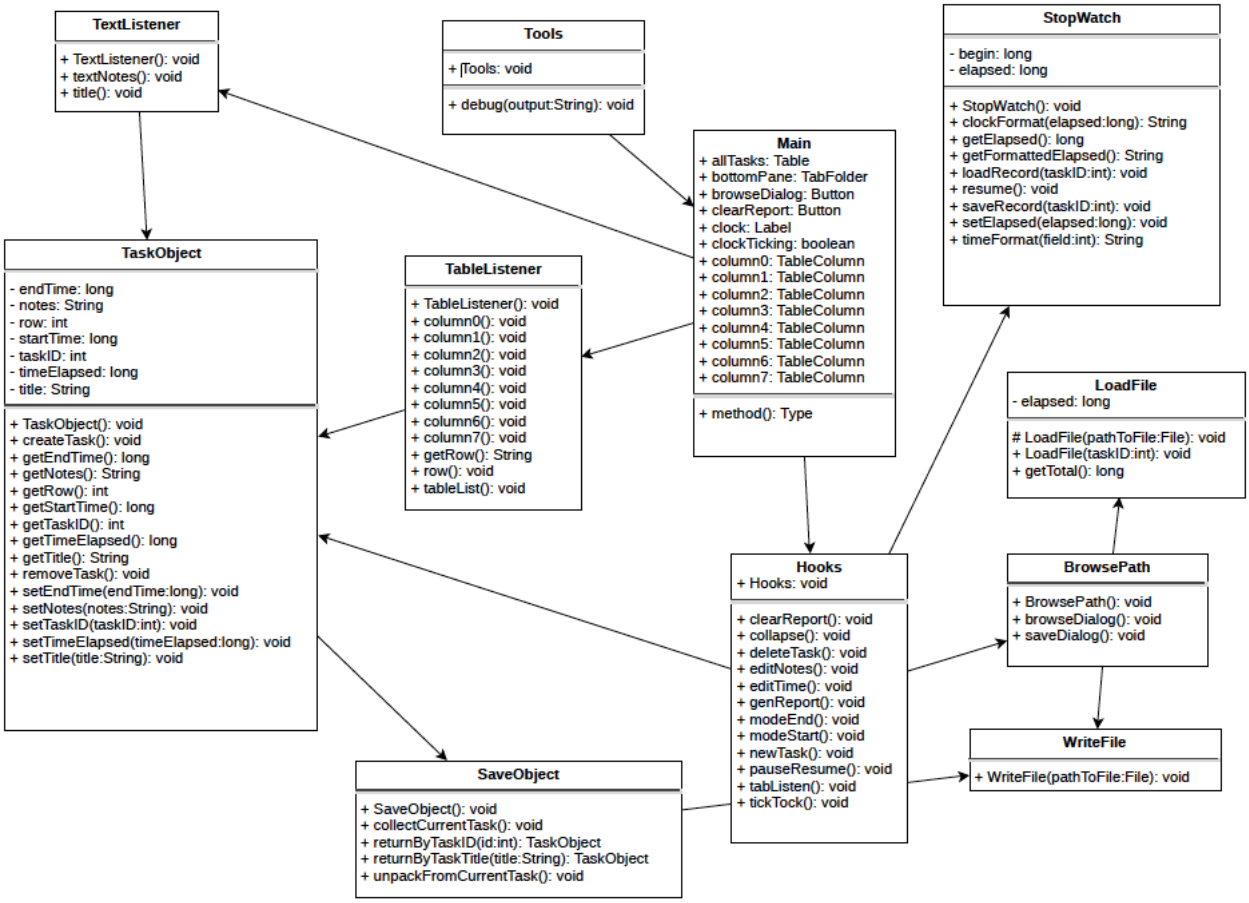


Use Case 12

Save As Task	Description
Actors	Field consultant, Configuration Table.
Description	A field consultant may save entire tasks which would include all of their data.
Data	File Name
Stimulus	User clicks the Save as button.
Response	Saves the entire as a .tsv file.
Comments	Saves as .tsv file.

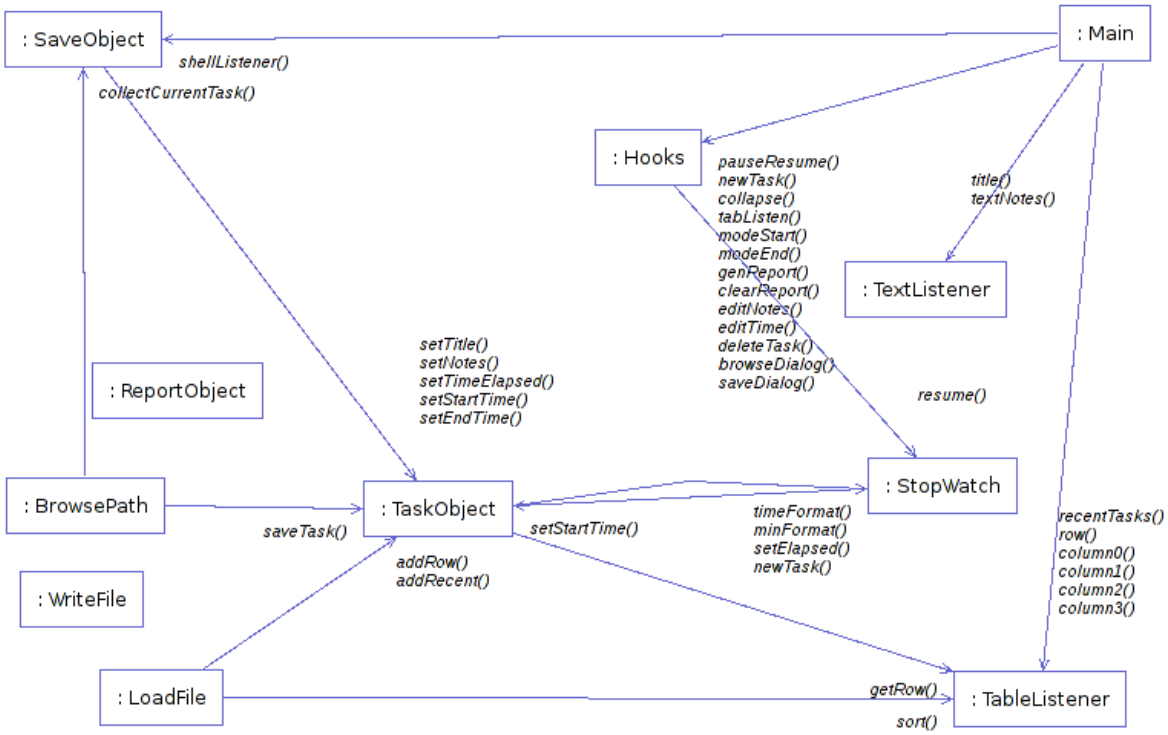
3.4 Classes / Objects

3.4.0.1 Class Overview



Class Diagram 1

3.4.0.2 Class Interaction



Class Diagram 2

3.4.1 BrowsePath

BrowsePath
+autoSave() : void +browseDialog() : void +saveDialog() : void

3.4.1.1 Functions

autoSave	Save table to loaded file
browseDialog	Loads file open selection in new window
saveDialog	Loads file save selection in new window

3.4.2 Hooks

Hooks
+browseDialog() : void +clearReport() : void +collapse() : void +deleteTask() : void +editNotes() : void +editTime() : void +genReport() : void +modeEnd() : void +modeStart() : void +newTask() : void +pauseResume() : void +saveDialog() : void +tabListen() : void +tickTock() : void

3.4.2.1 Functions

browseDialog	Listener for browse button
clearReport	Listener for clear button
collapse	Listener for collapse button
deleteTask	Listener for delete task button
editNotes	Listener for edit notes button
editTime	Listener for edit time button
genReport	Listener for generate report button
modeEnd	Listener for sort by end time radio button
modeStart	Listener for sort by start time radio button
newTask	Listener for new task button
pauseResume	Listener for pause/resume button
saveDialog	Listener for save as... button
tabListen	Listener for focused tab
tickTock	Toggle stopwatch start/stop thread

3.4.3 LoadFile

LoadFile
#LoadFile(pathToFile:String) : void

3.4.3.1 Functions

LoadFile	Constructor takes absolute path to filename (String) and reads TSV file into table

3.4.4 Main

Main	
+allTasks : Table +bottomPane : TabFolder +browseDialog : Button +cell : TableEditor +clearReport : Button +clock : Label +clockTicking : boolean +collapse : Button +column0 : TableColumn +column1 : TableColumn +column2 : TableColumn +column3 : TableColumn +column4 : TableColumn +column5 : TableColumn +column6 : TableColumn +column7 : TableColumn +column8 : TableColumn +configLoaded : boolean -ctrlAListener : Listener +currentTask : TaskObject	+deleteTask : Button +display : Display +down : Rectangle +editNotes : Button +editTime : Button +fileStatus : Label #frame : Shell +genReport : Button +inline : Text +interfaceDown : boolean +maxTaskID : int +modeEnd : Button +modeStart : Button +newTask : Button +pauseResume : Button +recentColumn0 : TableColumn +recentColumn1 : TableColumn +recentColumn2 : TableColumn +recentTasks : Table
#createContents() : void #initialize() : void +main(args:String[ ]) : void +open() : void +widgetDefaultSelected(e:SelectionEvent) : void +widgetSelected(e:SelectionEvent) : void	

3.4.4.1 Attributes

allTasks	Table that contains all task data
bottomPane	TabFolder that contains notes, management, reports and configuration tabs
browseDialog	Button that triggers load file dialog
cell	TableEditor that allows elapsed time to be modified
clearReport	Button that triggers empty report textbox
clock	Label that displays the running stopwatch HH:MM:SS.m (Hours, Minutes, Seconds, 1/10 of a Second)
clockTicking	Boolean used to edit pauseResume text, toggle StopWatch, and add elapsed to titlebar

collapse	Button that hides/shows bottomPane
column0	TableColumn that holds row number
column1	TableColumn that holds task title
column2	TableColumn that holds elapsed time (HH:MM:SS)
column3	TableColumn that holds recent status (+ / -)
column4	TableColumn that holds taskID (hidden)
column5	TableColumn that holds task notes (hidden)
column6	TableColumn that holds start time in epoch (hidden)
column7	TableColumn that holds end time in epoch (hidden)
column8	TableColumn that holds total time in milliseconds (hidden)
configLoaded	Boolean used to check if a file is loaded
ctrlAListener	Listener for <CTRL>+A hotkey for select-all in notes and report textboxes
currentTask	TaskObject that contains current task data
deleteTask	Button that deletes selected task from allTasks table
display	Display that shows the main GUI in the window
down	Rectangle that contains expanded window size
editNotes	Button that loads selected task from the table into top of recent tasks and the current task, pauses the timer and switches to Notes tab
editTime	Button that creates in-line cell editor for human-readable elapsed time
fileStatus	Label that displays file status
frame	Shell that holds main SWT window
genReport	Button that generates report into reports textbox
inline	Text that holds in-line cell editor text
interfaceDown	Boolean that holds expand/collapse status
maxTaskID	Integer that contains maximum taskID value
modeEnd	Radio button that changes reports sort order by end timestamps
modeStart	Radio button that changes reports sort order by start timestamps
newTask	Button that creates a new task and adds to recent list
pauseResume	Button that toggles stopwatch start/stop thread
recentColumn0	TableColumn that holds title
recentColumn1	TableColumn that holds elapsed time (HH:MM:SS)
recentColumn2	TableColumn that holds taskID (hidden)
recentTasks	Table that holds last 4 recently used tasks
reportToggle	Boolean that holds report sort order
saveDialog	Button that triggers file save dialog
selectedFile	String that contains absolute path to loaded file
sort0	Boolean that contains row# sort order
sort1	Boolean that contains title sort order
sort2	Boolean that contains elapsed sort order

sort3	Boolean that contains recent status sort order
tab1	Notes TabItem that contains title and notes textboxes
tab2	Management TabItem that contains all tasks, edit notes, edit time and delete tasks buttons
tab3	Reports TabItem that contains reports textbox, sort radio buttons, generate and clear buttons
tab4	Configuration TabItem that contains fileStatus label, textDir textbox, browse and save as buttons
textDir	Textbox that contains absolute path to loaded file
textNotes	Styled Textbox that contains task notes
textReport	Textbox that contains generated reports
title	Textbox that contains task title
untitled	Integer that contains Untitled-# for new tasks
up	Rectangle that contains collapsed window size

3.4.4.2 Functions

createContents	Constructor that creates, stylizes and places SWT widgets in window frame
intialize	Constructor that initializes variables on start
main	Creates a new window shell
open	Opens the newly created shell
widgetDefaultSelected	SWT default hooks
widgetSelected	SWT override hooks

3.4.5 ReportObject

ReportObject
-dateFormat(epoch : String) : String -getData(reports : String [ ] [ ]) : void +newReport() : void -printReport(reports : String [ ] [ ]) : void -sortReport(reports : String [ ] [ ], sortMode : int) : void

3.4.5.1 Functions

dateFormat	Converts milliseconds from epoch into dd/mm/yy HH:MM:SS
getData	Get data from table
newReport	Creates new report from sorted data
printReport	Outputs formatted report to report textbox
sortReport	Sorts task data by start or end time

3.4.6 SaveObject

SaveObject
+collectCurrentTask() : void +shellListener() : void

3.4.6.1 Functions

collectCurrentTask	Gather data from current task to prepare to save in table
shellListenenr	Listener that triggers close dialog on exit

3.4.7 StopWatch

StopWatch
+begin : long -change : long +elapsed : long
+clearTimer() : void +clockFormat(elapsed : long) : String +countChange() : void +getElapsed() : long +getFormattedElapsed() : String +minFormat(elapsed : long) : String +newTask() : void +resume() : void +setElapsed(elapsed : long) : void +timeFormat(field : int) : String

3.4.7.1 Attributes

begin	Long that contains start value for stopwatch
change	Long that contains interval up to 1 second
elapsed	Long that contains elapsed value for stopwatch

3.4.7.2 Functions

clearTimer	Resets timer
clockFormat	Formats long elapsed and returns String
countChange	Updates title once a second
getElapsed	Returns long elapsed time
getFormattedElapsed	Returns formatted (HH:MM:SS.m) time String
minFormat	Returns formatted (HH:MM:SS) time String
newTask	Creates new stopwatch thread
resume	Resumes stopwatch background thread
setElapsed	Set elapsed time as stopwatch start time
timeFormat	Formats integer as double-digit String

3.4.8 TableListener

TableListener
+column0() : void +column1() : void +column2() : void +column3() : void +getRow() : int +recentTasks() : void +row() : void +sort(order : boolean, collnt : int) : void

3.4.8.1 Functions

column0	Listener for row# header
column1	Listener for title header
column2	Listener for elapsed header
column3	Listener for recent status header
getRow	Returns selected row integer
recentTasks	Save task to recent tasks
row	Listener for rows
sort	Sorts table in order/reversed order by selected column

3.4.9 TaskObject

TaskObject
-elapsed : String -endTime : long -notes : String -startTime : long -taskID : int -title : String -total : long
+TaskObject() : void +addRecent(newID : int, list : String [ ]) : void +addRow(newID : int, list : String [ ]) : void +checkRecent(selected : int) : int +checkTable(selected : int) : int +createTask() : void +getElapsed() : String +getEndTime() : long +getNotes() : String +getStartTime() : long +getTaskID() : int +getTitle() : String +getTotal() : long +newTask() : void +removeTask(row : int) : void +returnTaskFromIndex(rowindex : int) : TaskObject +saveCurrentToRow() : void +saveTask(taskToSave : TaskObject) : void +saveTaskToRow(ID : int) : void +searchRecentByID(ID : int) : int +searchTableByID(ID : int) : int +setElapsed(elapsed : String) : void +setEndTime(endTime : long) : void +setNotes(notes : String) : void +setStartTime(startTime : long) : void +setTaskID(taskID : int) : void +setTitle(title : String) : void +setTotal(total : long) : void +unpackFromCurrentTasktoFields(taskToUnpack : TaskObject) : void

3.4.9.1 Attributes

elapsed	String that contains elapsed time (HH:MM:SS)
endTime	Long that contains end time in epoch
notes	String that contains notes
startTime	Long that contains start time in epoch
taskID	Integer that contains taskID
title	String that contains title
total	Long that contains total elapsed time

3.4.9.1 Functions

addRecent	Adds new task to top of recent list
addRow	Adds new row to bottom of table
checkRecent	Returns index of task if present in recent task list
checkTable	Returns index of task if present in all task list
createTask	Creates task
getElapsed	Returns elapsed time (HH:MM:SS)
getEndTime	Returns end time in epoch
getNotes	Returns notes
getStartTime	Returns start time in epoch
getTaskID	Returns taskID
getTitle	Returns title
getTotal	Returns total elapsed time
newTask	New task
removeTask	Deletes task from table and recent list
returnTaskFromIndex	Returns task data from given index
saveCurrentToRow	Save current task
saveTask	Save task
saveTaskToRow	Update current task to table and recent list
searchRecentbyID	Returns recent index of task if present by taskID
searchTableByID	Returns table index of task if present by taskID
setElapsed	Sets elapsed time of current task
setEndTime	Sets end time of current task
setNotes	Sets notes for current task
setStartTime	Sets start time of current task
setTaskID	Sets taskID of current task
setTitle	Sets title of current task
setTotal	Sets total time of current task
UnpackFrom CurrentTaskToFields	Populates current task by task from table

3.4.10 TextListener

TextListener
+textNotes() : void +title() : void

3.4.10.1 Functions

textNotes	Update task notes to table and recent list when edited
title	Update task title to table and recent list when edited

3.4.11 Tools

Tools
+debug(output : String) : void

3.4.11.1 Functions

debug Output diagnostic information to console

3.4.12 WriteFile

WriteFile
#WriteFile(pathToFile : String) : void

3.4.12.1 Functions

WriteFile	Constructor takes absolute path to filename (String) and write table to TSV file

3.5 Non-Functional Requirements

3.5.1 Performance Requirements

The program must run with the allotted amount of memory and processing power available in the machine and should take steps to avoid using more resources than are required. For this reason the background thread has a 100 millisecond sleep between updating the clock to free up resources. It must handle user interaction without a human noticeable delay to maintain a proper use workflow.

3.5.2 Reliability

TimeMe was designed for use with billable hours in mind. Many employers incorporate 15 minute interval accuracy into billable hours and for this reason, TimeMe is accurate up to the second. The back-end relies on the OS system clock (which may either get its time from the hardware clock or a network time proxy) to the millisecond displaying accuracy up to a tenth of a second. Additionally rounding may occur up to the second due to editing the elapsed interval manually.

Reports generated are sorted by Epoch timestamps ensuring the utmost accuracy available. However as with any clock-dependent software its accuracy relies on the host OS clock.

The program must close gracefully and every effort has been made to prevent infinite loops and race cases. This reliability is affected by the Java Virtual Machine runtime and the host Operating System.

3.5.3 Availability

TimeMe is a stand-alone local application, capable of running from a removable drive. Its availability is directly related to the availability of the Java runtime. Additionally SWT requires a compatible CPU architecture and OS widget toolkit. All data is stored in memory until the user saves to a file.

3.5.4 Security Requirements

TimeMe does not hold any private user data within the application. All users are anonymous and the data is stored in cleartext in a TSV formatted text file. It is up to the user to secure the configuration file as portability was a higher priority. As the outputted file is a standard TSV file it can be imported into spreadsheet software such as Microsoft Excel, LibreOffice Calc and Google Spreadsheets.

3.5.5 Maintainability

TimeMe has very few dependencies on third party systems. Maintenance will be necessary for new versions of the JRE and SWT. However, these dependencies would normally be taken care of through the user's normal OS updates. Because the code does not rely on third-party systems, TimeMe will not need iterative maintenance. TimeMe will only need to be maintained in order to resolve bugs reported by users.

3.5.6 Portability

TimeMe is a portable application meant to be used on the go and at various workstations. The application should be small enough to fit on a thumb drive and if required to run from it. The

platform-specific SWT library dependencies are bundled within the single cross-platform, executable JAR file.

3.6 Inverse Requirements

Left Intentionally Blank

3.7 Design Constraints

The program must run on the presentation workstation running Windows 7 x86_64.
It must either be portable or compiled on the workstation which limits the programming languages available. For this reason Java was chosen as the JRE comes pre-installed in the environment and produces portable JAR executable files. Additionally the Java API libraries must either be standard or come bundled with the executable and SWT was ultimately chosen for this reason as well as its native OS widgets.

3.8 Logical Database Requirements

Left Intentionally Blank

3.9 Other Requirements

Code development environment:

- Eclipse IDE
- egit (Eclipse Team add-on)
- Google Code issue (bug tracker) connector (Eclipse Mylyn connector)
- Google Code project hosting
- git repository (distributed version control system)

More tools:

- GIMP (mockups and icon)
- Archbang/Archlinux usb install

Documentation:

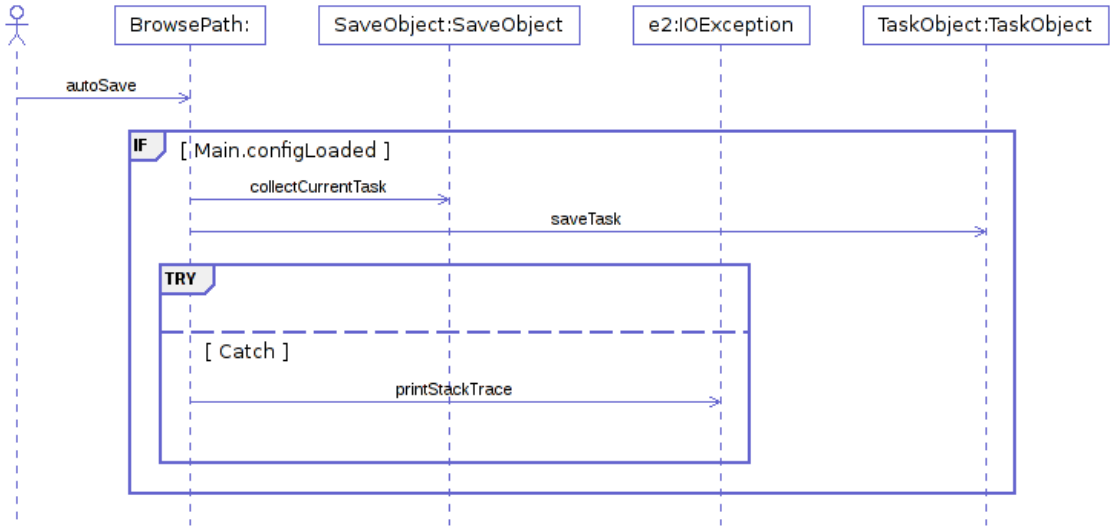
- Google Docs
- Microsoft Word 2010
- Microsoft Visio (Activity diagrams)
- Javadoc as UML diagram plugin for Eclipse (UML diagrams)
- Modelgoon plugin for Eclipse (Sequence diagrams, Interaction diagram)
- Diagram.ly (State diagrams)
- Pencil add-on for Firefox
- Imagemagick
- Ghostscript
- BASH scripts

4. Analysis Models

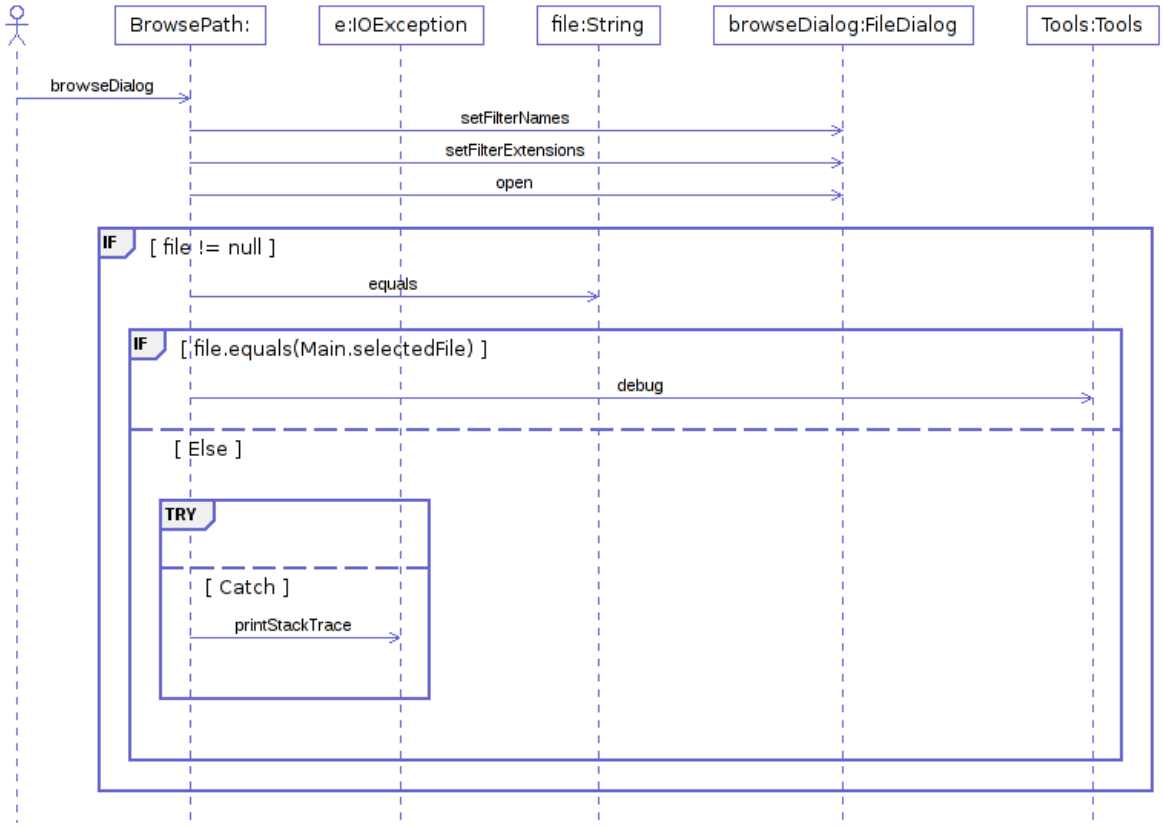
4.1 Sequence Diagrams

4.1.1 BrowsePath

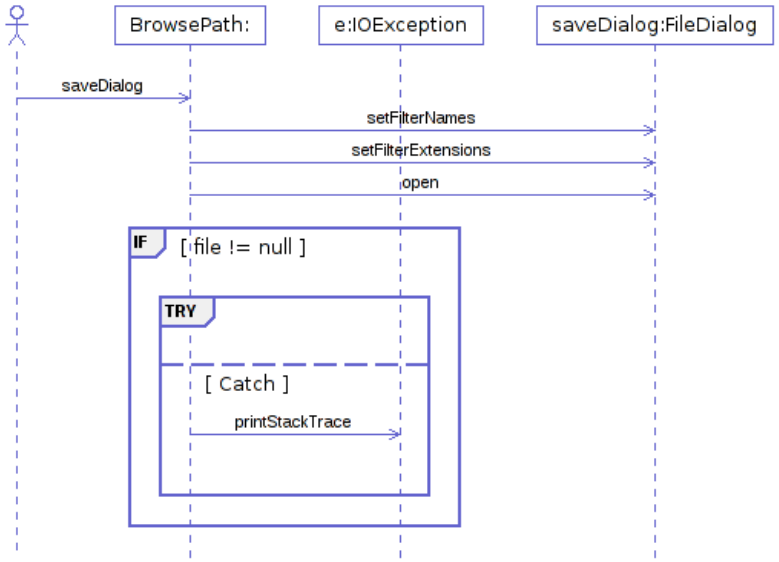
4.1.1.1 BrowsePath—AutoSave



4.1.1.2 BrowsePath—BrowseDialog



4.1.1.3 BrowserPath—SaveDialog

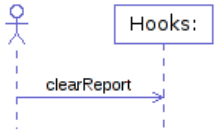


4.1.2 Hooks

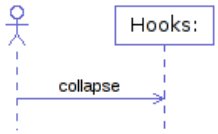
Class contains button listeners that trigger functions in other classes.

4.1.2.1 Hooks—BrowseDialog

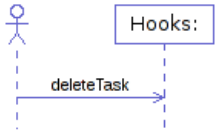
4.1.2.2 Hooks—ClearReport



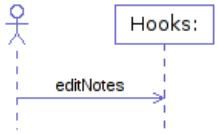
4.1.2.3 Hooks—Collapse



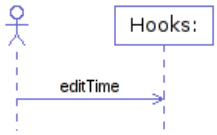
4.1.2.4 Hooks—DeleteTask



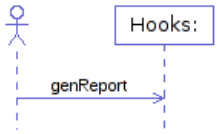
4.1.2.5 Hooks—EditNotes



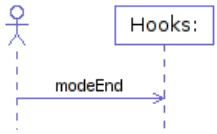
4.1.2.5 Hooks—EditTime



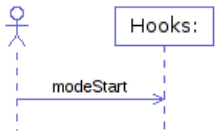
4.1.2.6 Hooks—GenReport



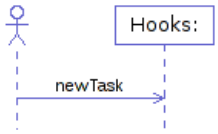
4.1.2.7 Hooks—ModeEnd



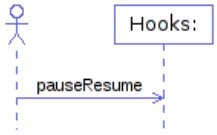
4.1.2.8 Hooks—ModeStart



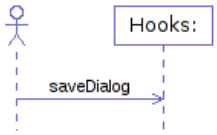
4.1.2.9 Hooks—NewTask



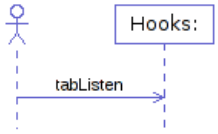
4.1.2.10 Hooks—PauseResume



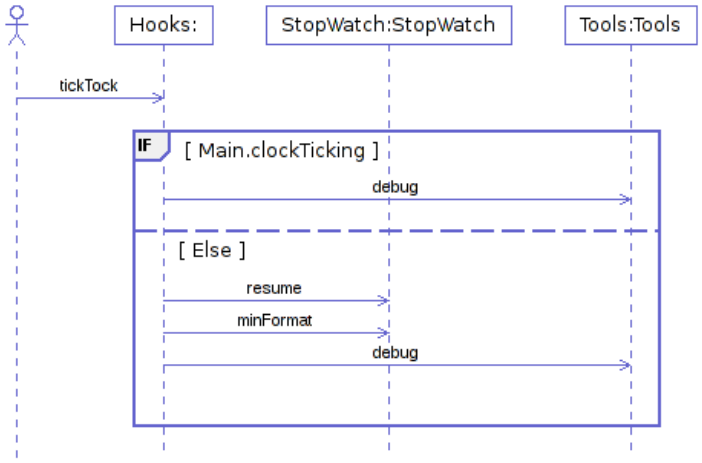
4.1.2.11 Hooks—SaveDialog



4.1.2.12 Hooks—TabListen

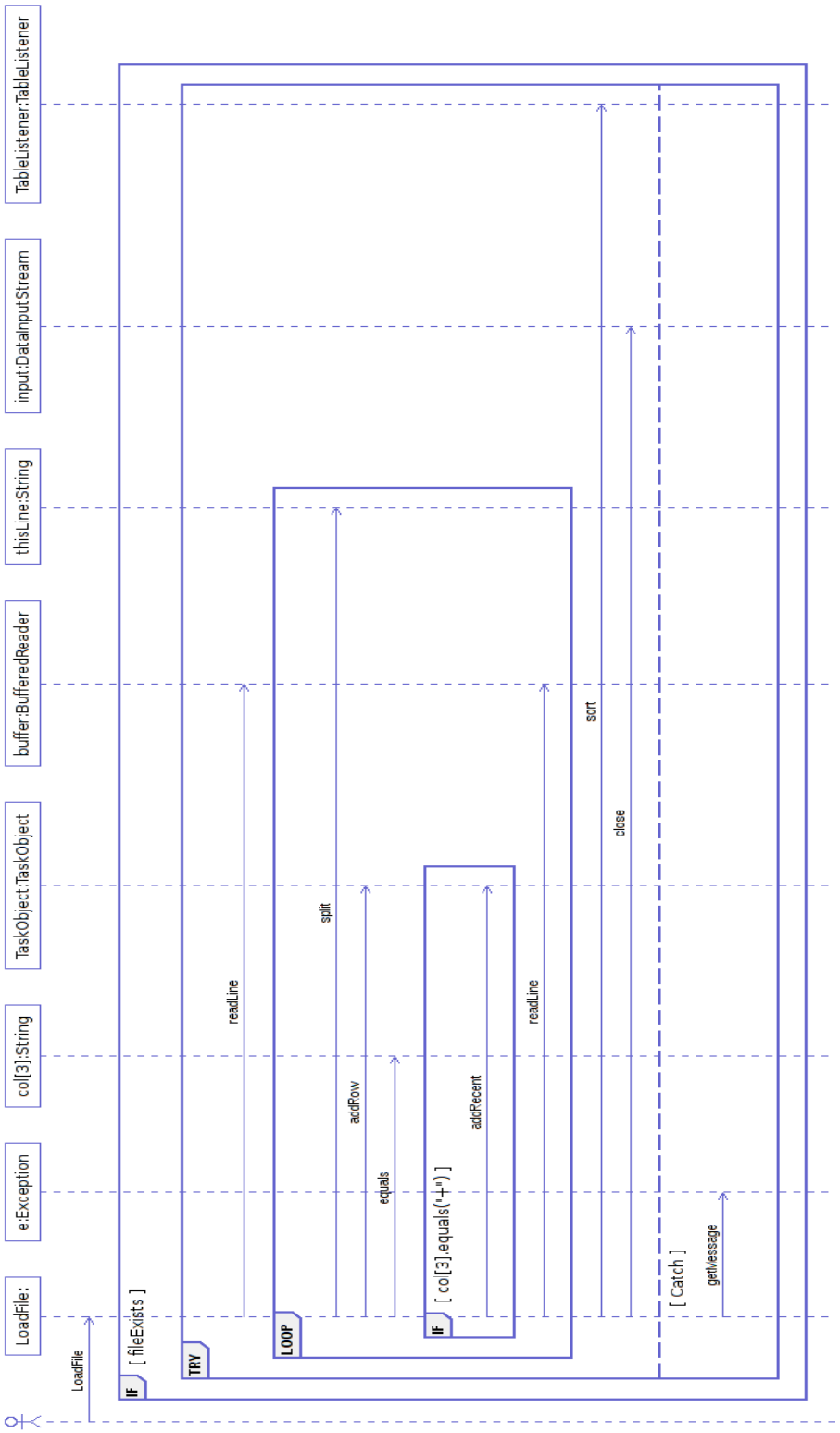


4.1.2.13 Hooks—TickTock



4.1.3 LoadFile

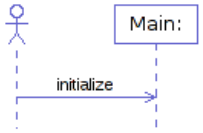
4.1.3.1 LoadFile—LoadFile



4.1.4 Main

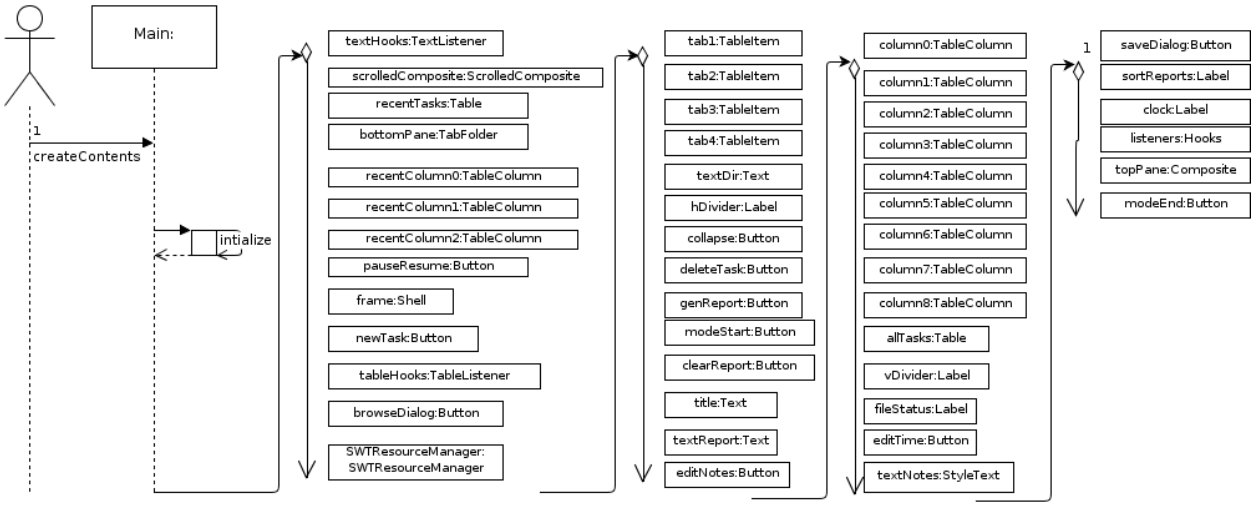
Function is requirement of SWT standards

4.1.4.1 Main—Initialize

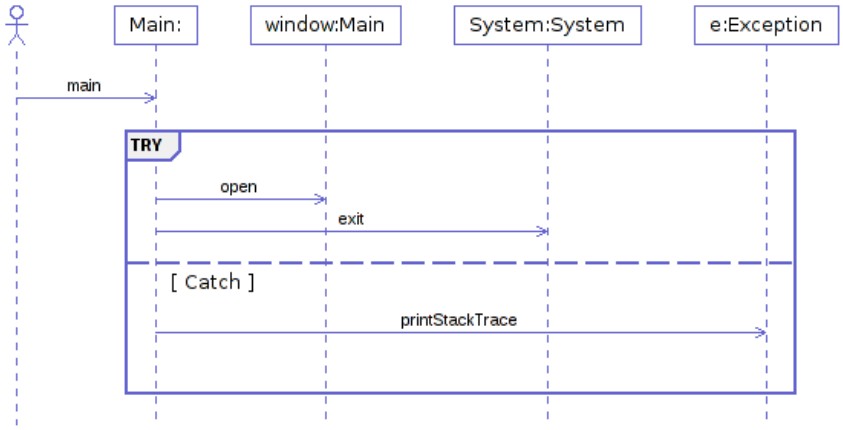


4.1.4.2 Main—CreateContents

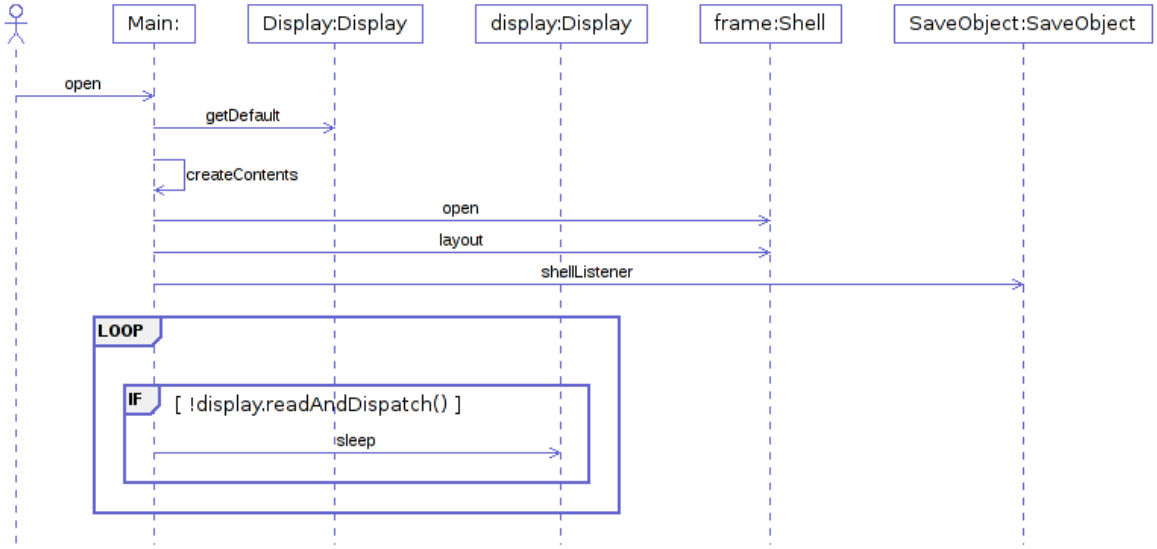
Diagram layout designed to fit constraints of page size



4.1.4.3 Main—Main

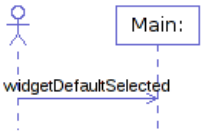


4.1.4.4 Main—Open



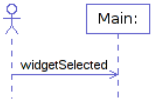
Function is requirement of SWT standards

4.1.4.5 Main—WidgetDefaultSelected



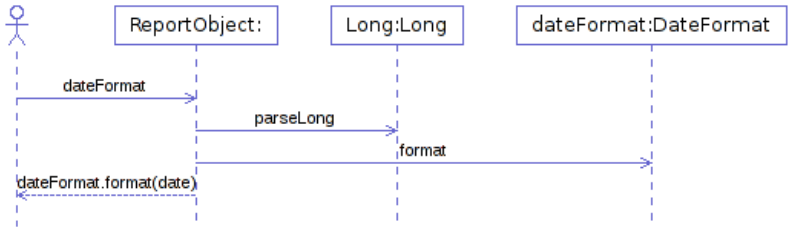
Function is requirement of SWT standards

4.1.4.6 Main—WidgetSelected

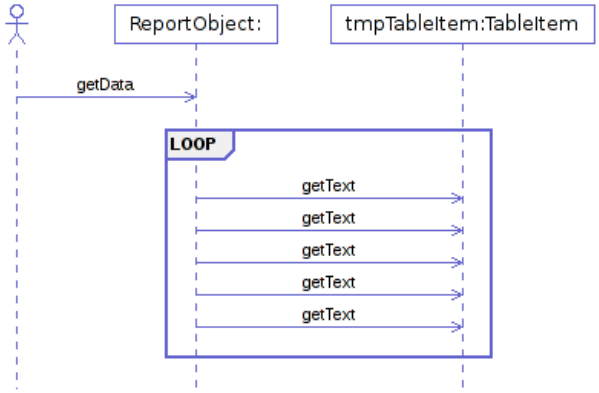


4.1.5 ReportObject

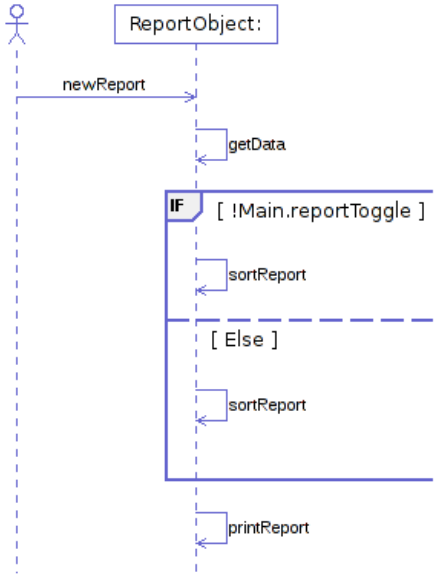
4.1.5.1 ReportObject—DateFormat



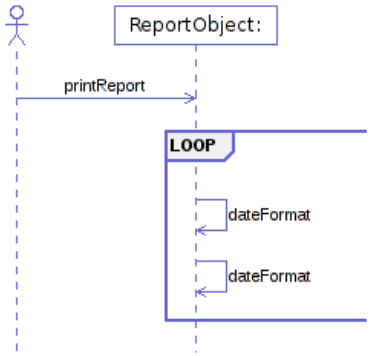
4.1.5.2 ReportObject—GetData



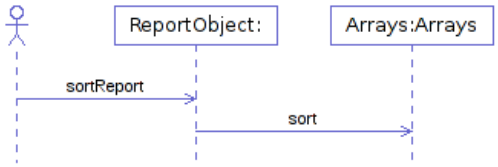
4.1.5.3 ReportObject—NewReport



4.1.5.4 ReportObject—PrintReport

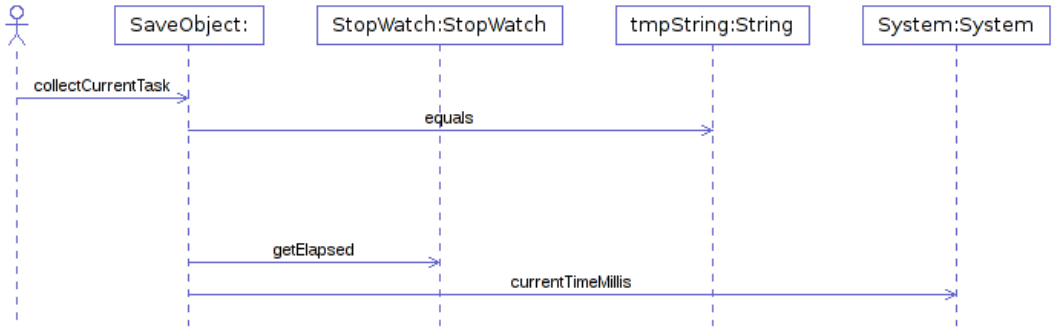


4.1.5.5 ReportObject—SortReport



4.1.6 SaveObject

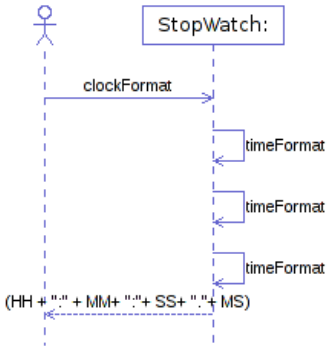
4.1.6.1 SaveObject—CollectCurrentTask



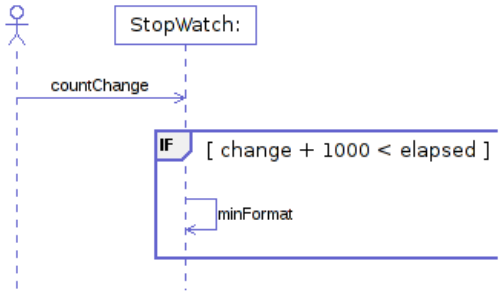
4.1.6.2 SaveObject—ShellListener

4.1.7 StopWatch

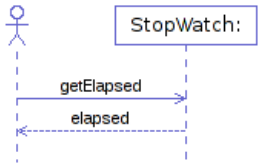
4.1.7.1 StopWatch—ClockFormat



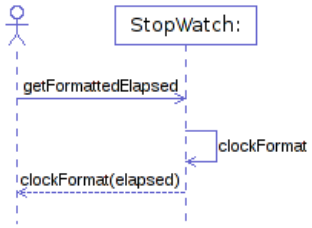
4.1.7.2 StopWatch—CountChange



4.1.7.3 StopWatch—GetElapsed

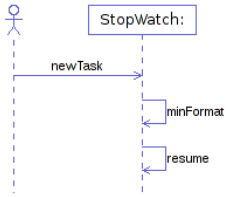


4.1.7.4 StopWatch—GetFormattedElapsed

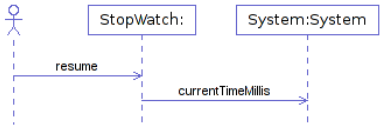


4.1.7.5 StopWatch—MinFormat

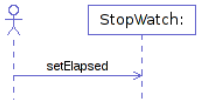
4.1.7.6 StopWatch—NewTask



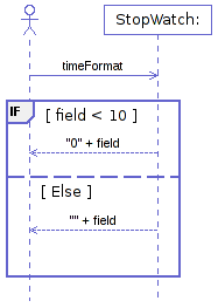
4.1.7.7 StopWatch—Resume



4.1.7.8 StopWatch—SetElapsed



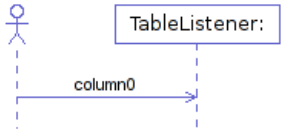
4.1.7.9 StopWatch—TimeFormat



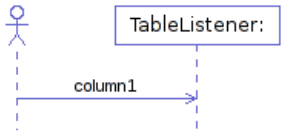
4.1.8 TableListener

Class contains button listeners that trigger functions in other classes.

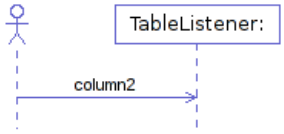
4.1.8.1 TableListener—Column0



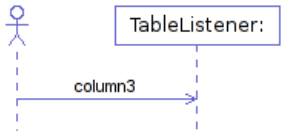
4.1.8.2 TableListener—Column1



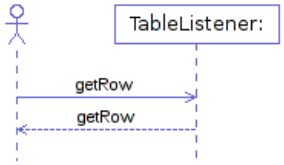
4.1.8.3 TableListener—Column2



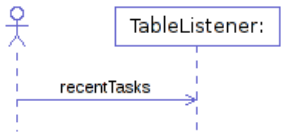
4.1.8.4 TableListener—Column3



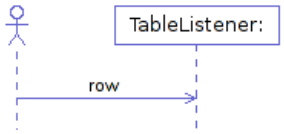
4.1.8.5 TableListener—GetRow



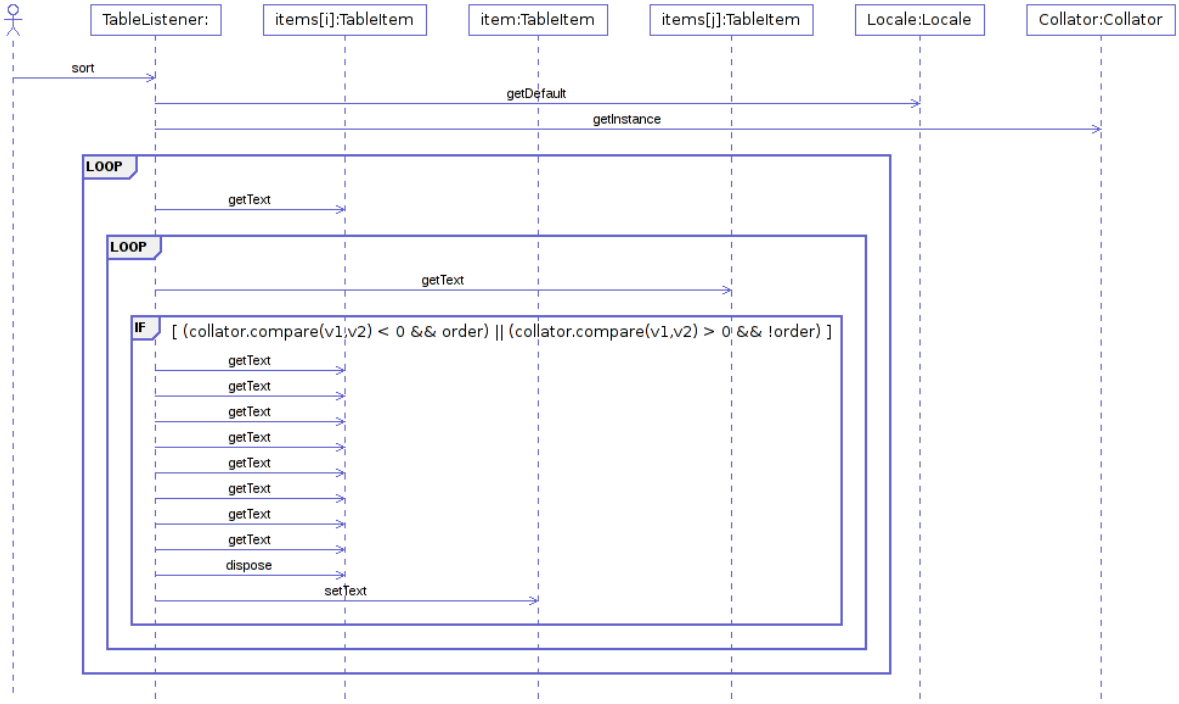
4.1.8.6 TableListener—RecentTasks



4.1.8.7 TableListener—Row



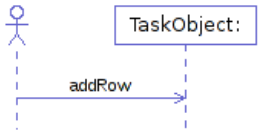
4.1.8.8 TableListener—Sort



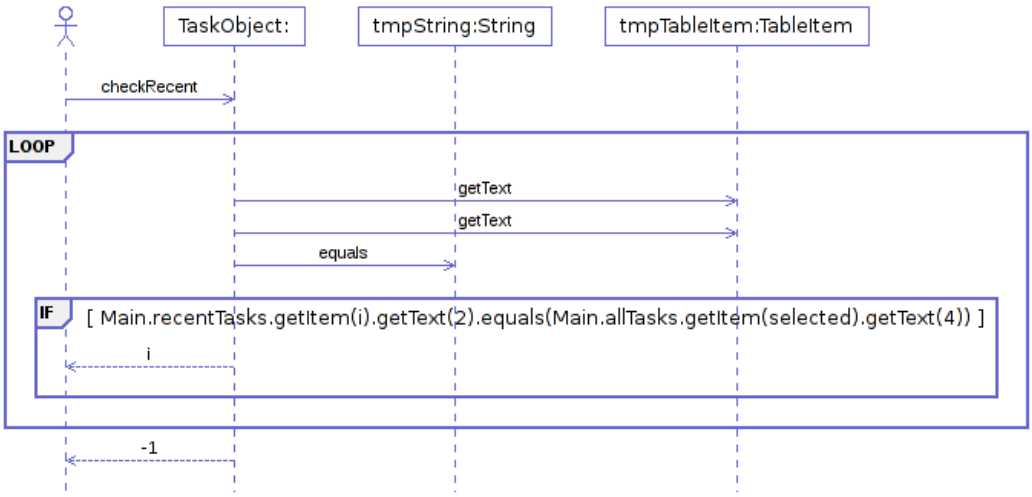
4.1.9 TaskObject

4.1.9.1 TaskObject—AddRecent

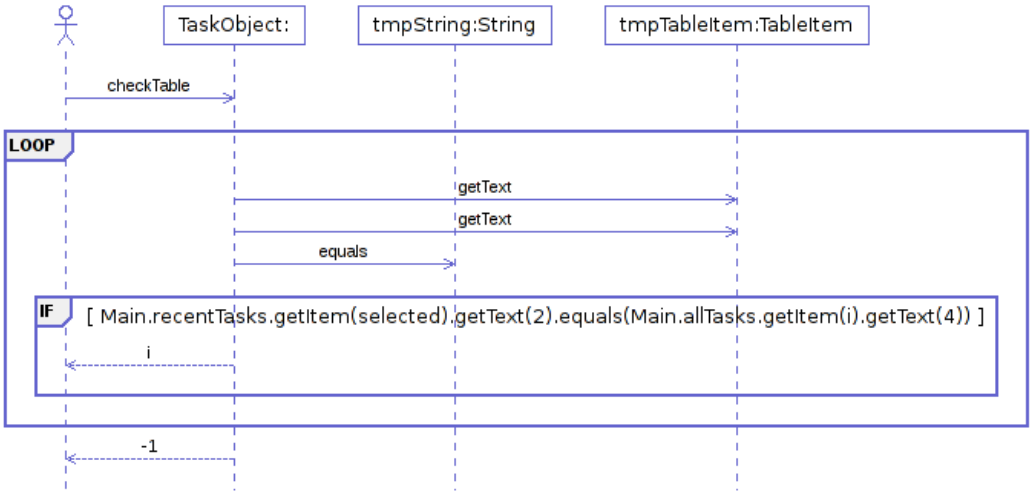
4.1.9.2 TaskObject—AddRow



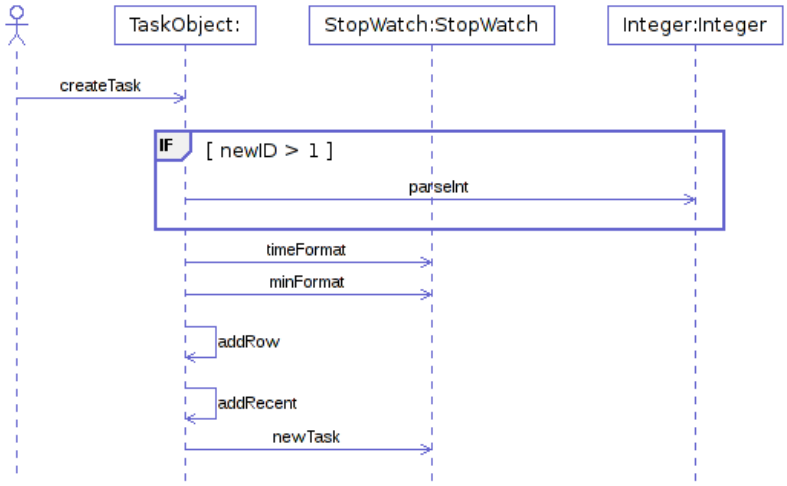
4.1.9.3 TaskObject—CheckRecent



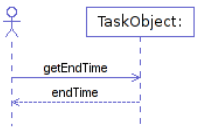
4.1.9.4 TaskObject—CheckTable



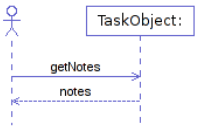
4.1.9.5 TaskObject—CreateTask



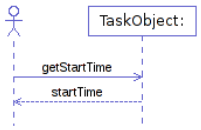
4.1.9.6 TaskObject—GetEndTime



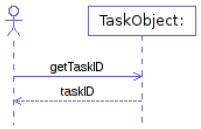
4.1.9.7 TaskObject—GetNotes



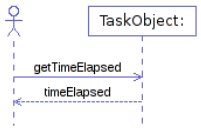
4.1.9.8 TaskObject—GetStartTime



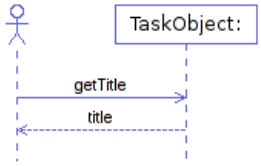
4.1.9.9 TaskObject—GetTaskID



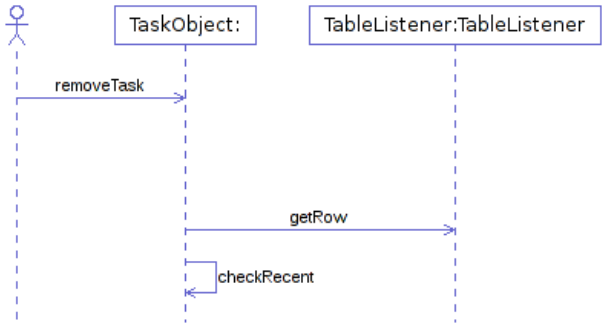
4.1.9.10 TaskObject—GetTimeElapsed



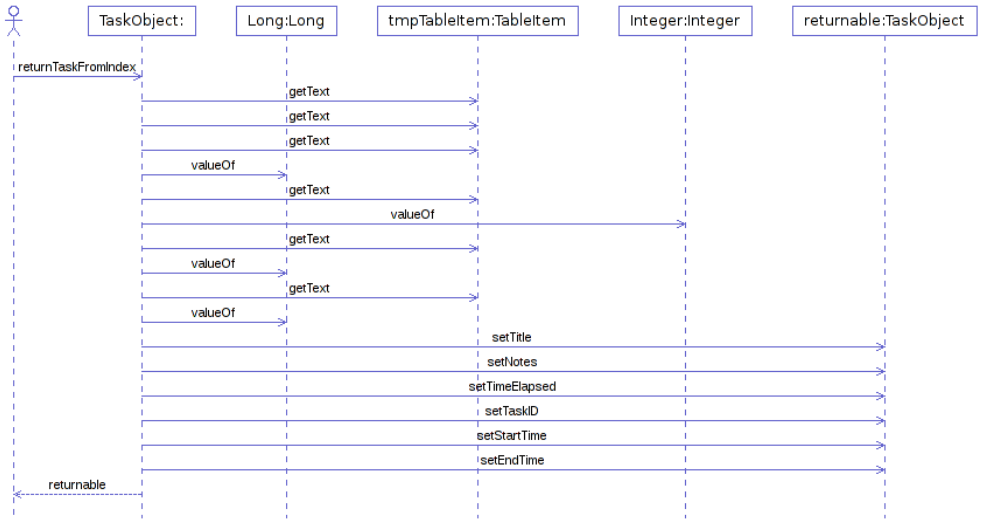
4.1.9.11 TaskObject—GetTitle



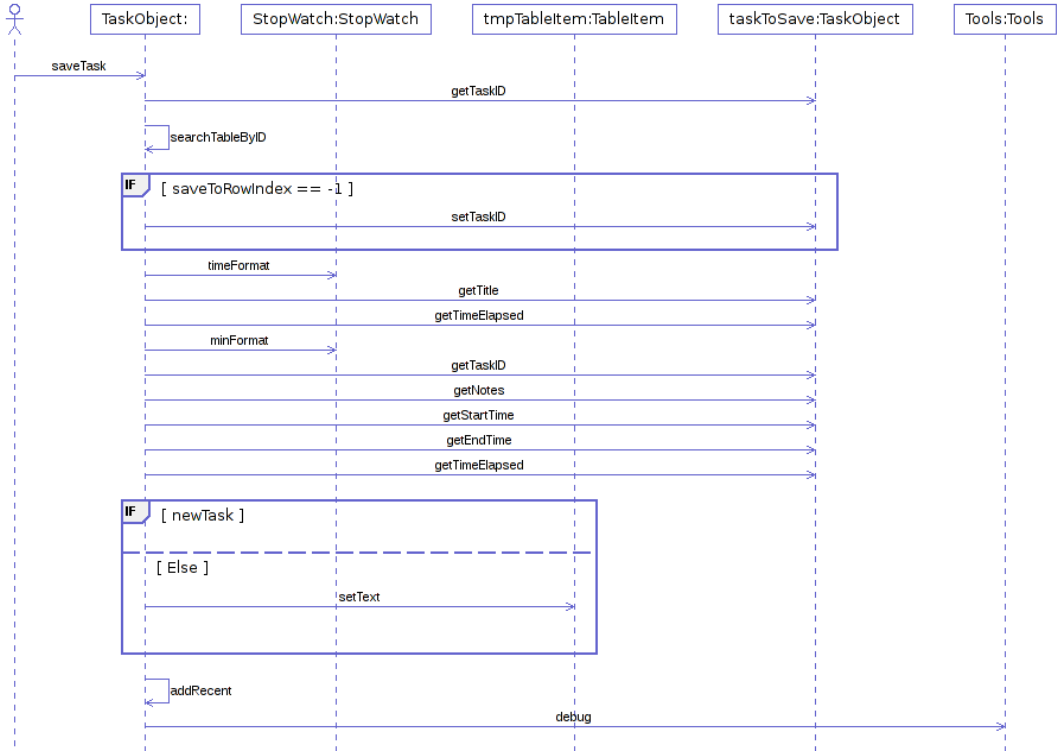
4.1.9.12 TaskObject—RemoveTask



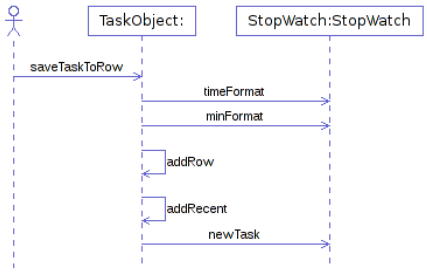
4.1.9.13 TaskObject—ReturnTaskFromIndex



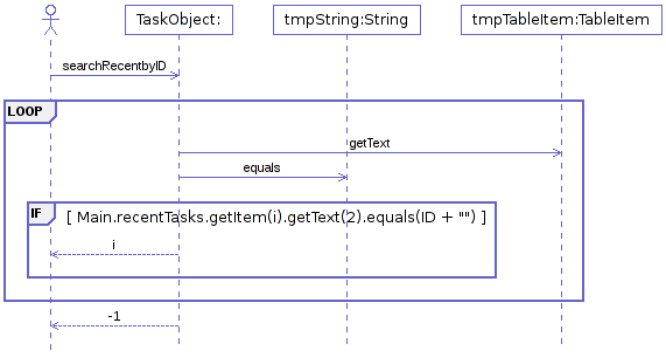
4.1.9.14 TaskObject—SaveTask



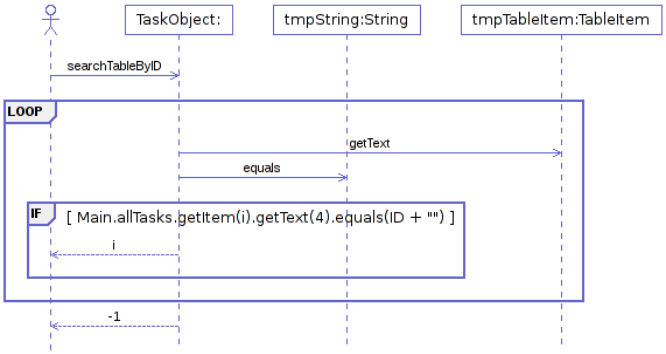
4.1.9.15 TaskObject—SaveTaskToRow



4.1.9.16 TaskObject—SearchRecentByID

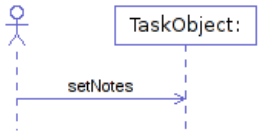


4.1.9.17 TaskObject—SearchTableByID

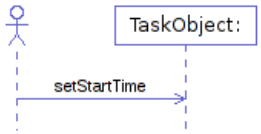


4.1.9.18 TaskObject—SetEndTime

4.1.9.19 TaskObject—SetNotes



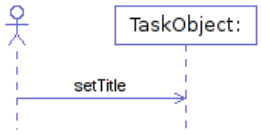
4.1.9.20 TaskObject—SetStartTime



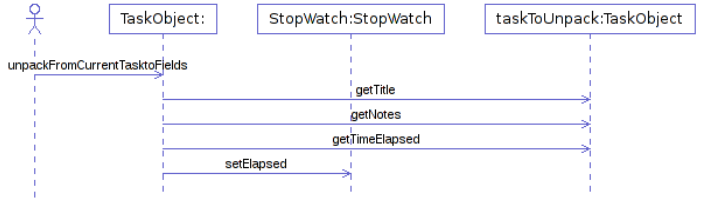
4.1.9.21 TaskObject—SetTaskID

4.1.9.22 TaskObject—SetTimeElapsed

4.1.9.23 TaskObject—SetTitle



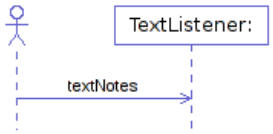
4.1.9.24 taskObject—UnpackFromCurrentTaskToFields



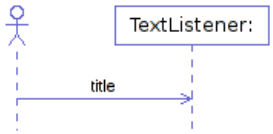
4.1.10 TextListener

Class contains textbox listeners that trigger functions in other classes.

4.1.10.1 TextListener—TextNotes



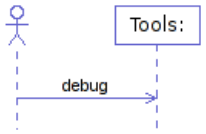
4.1.10.2 TextListener—Title



4.1.11 Tools

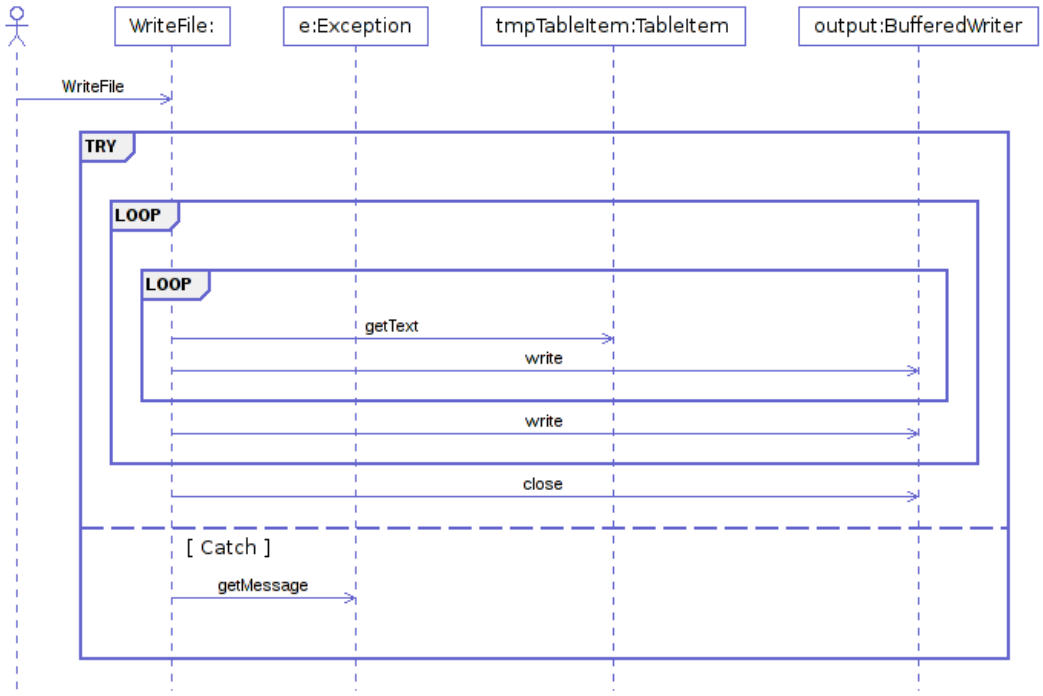
Class contains function to print debug output to console

4.1.11.1 Tools—Debug



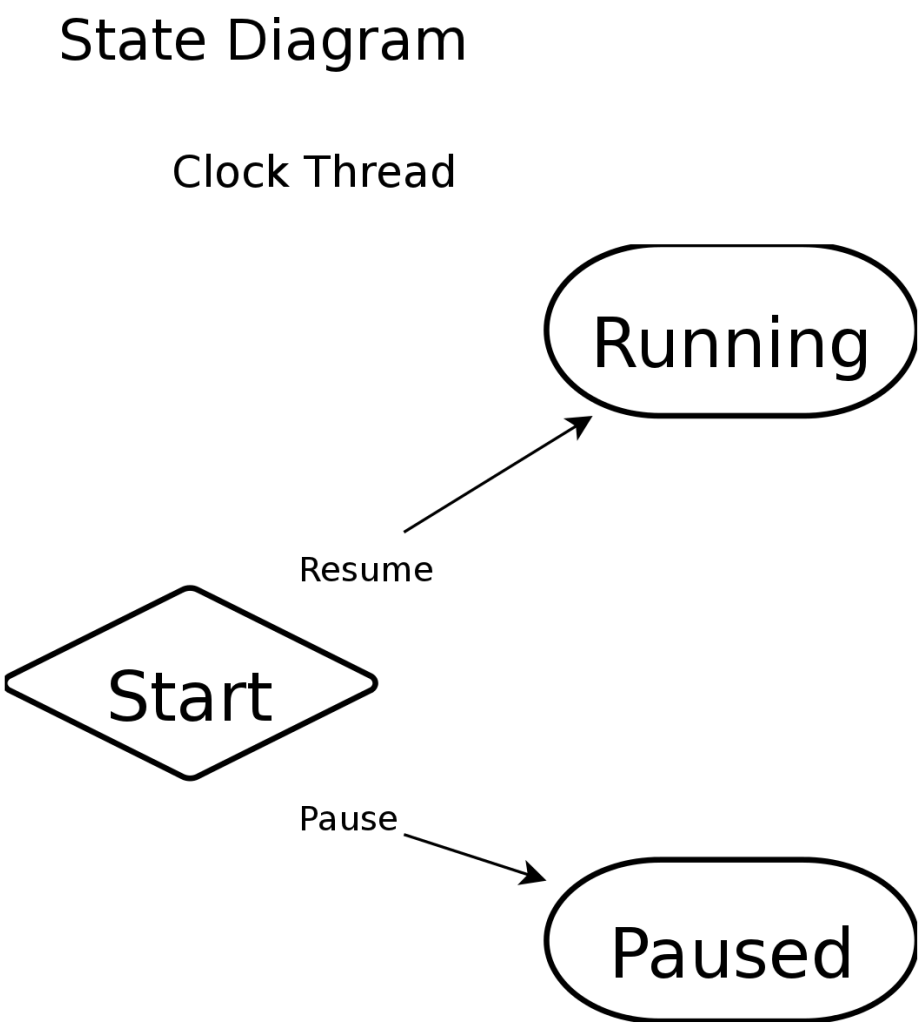
4.1.12 WriteFile

4.1.12.1 WriteFile—WriteFile

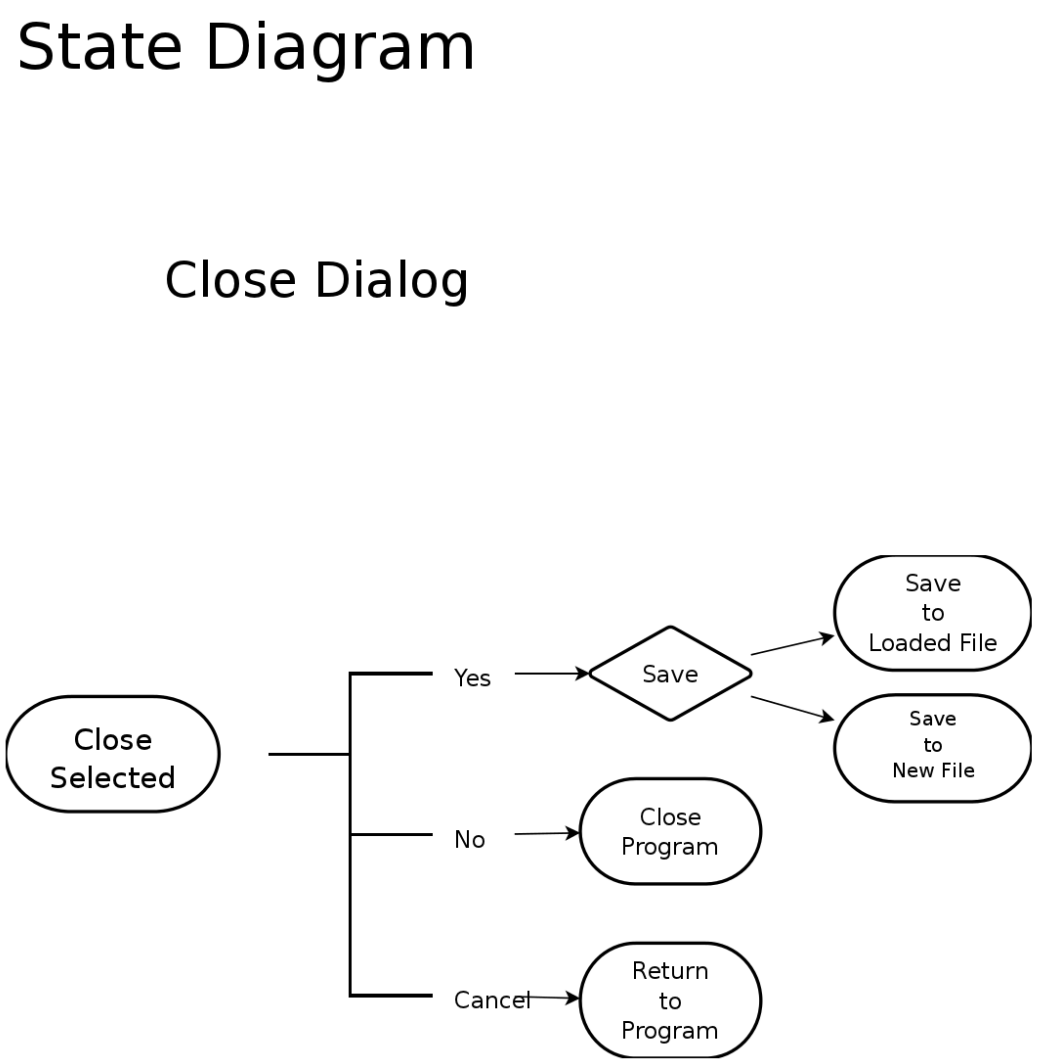


4.2 State-Transition Diagrams (STD)

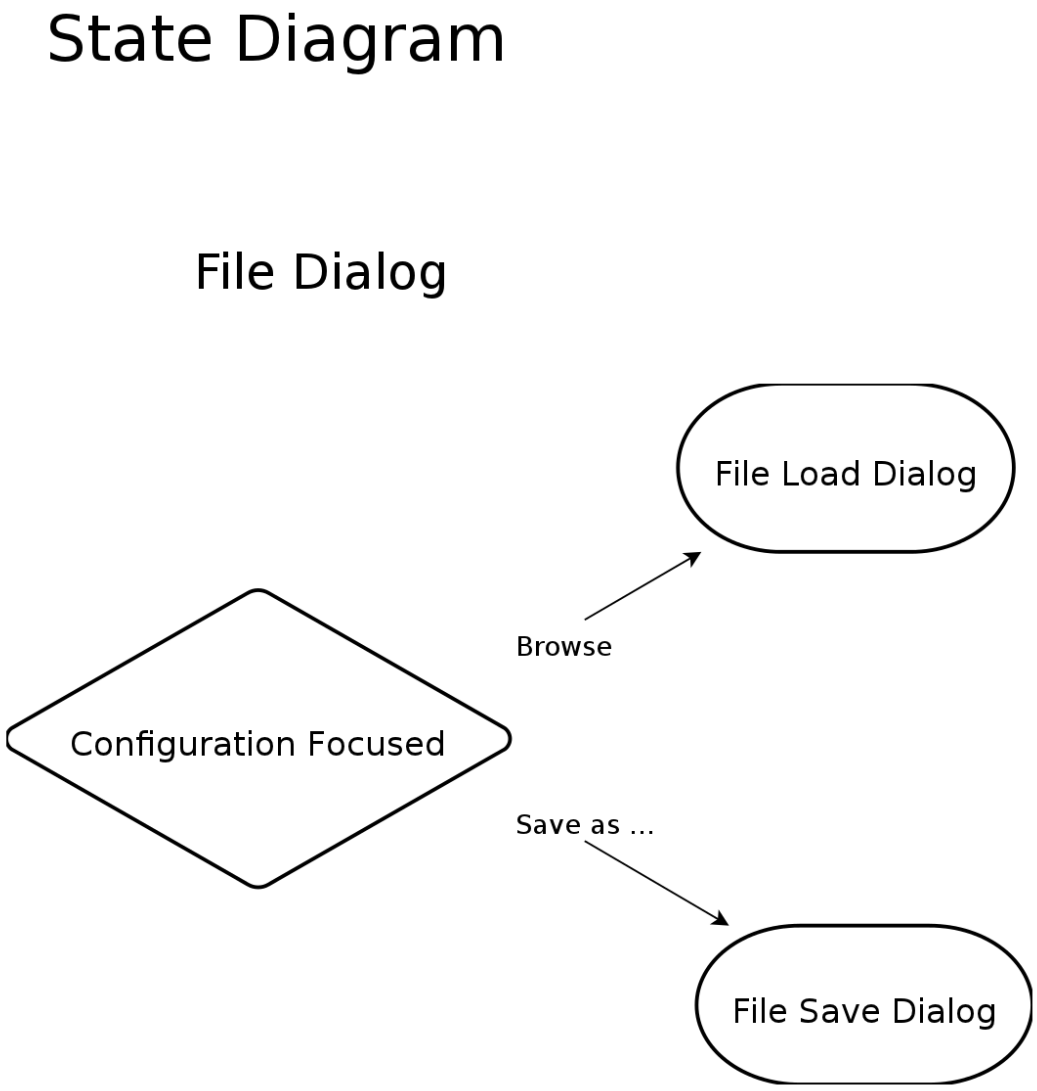
4.2.1 Clock State Diagram



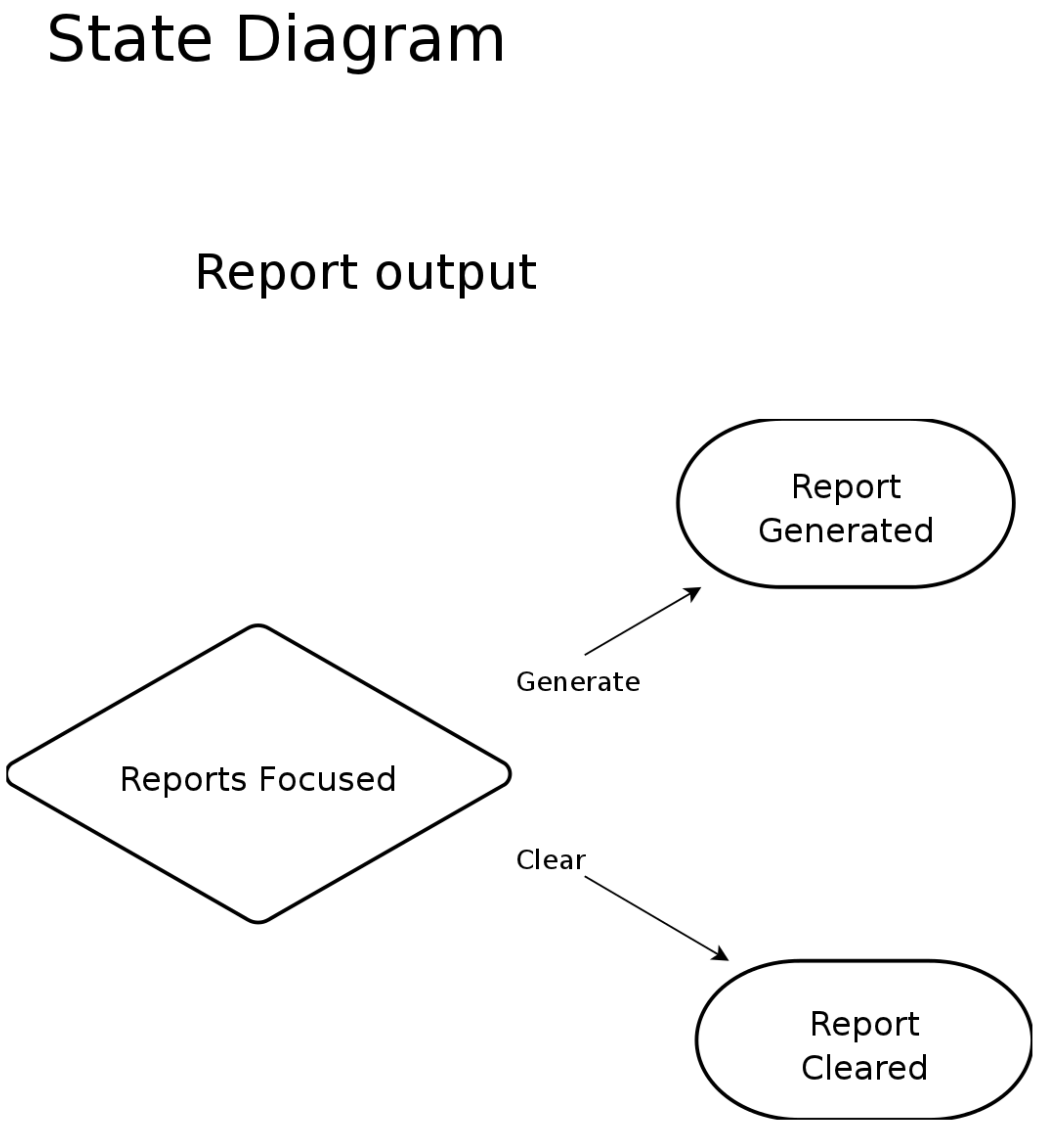
4.2.2 Close State Diagram



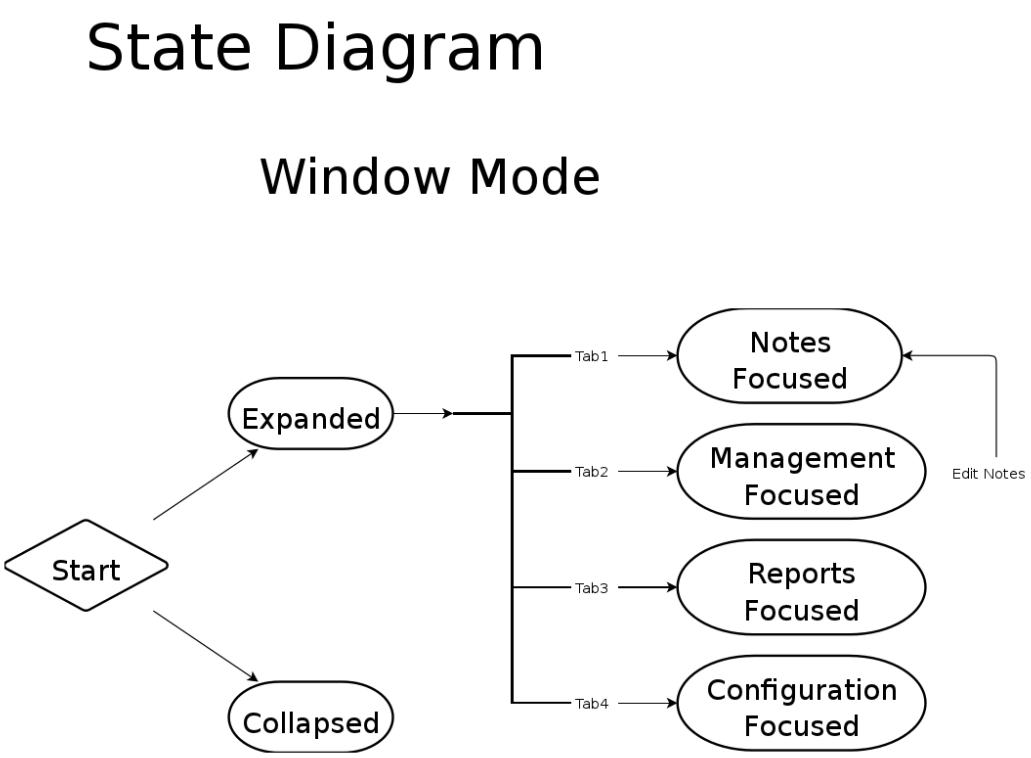
4.2.3 File State Diagram



4.2.4 Reports State Diagram

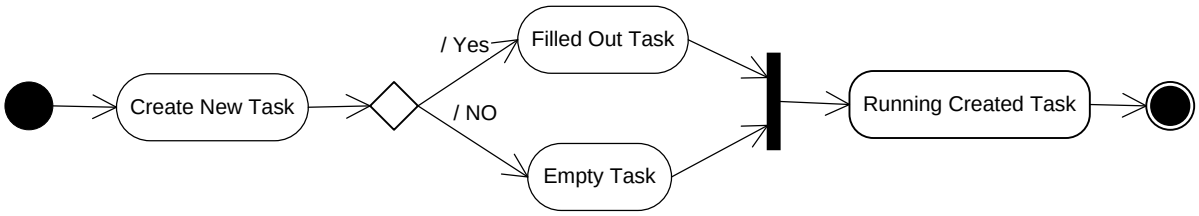


4.2.5 User Interface State Diagram



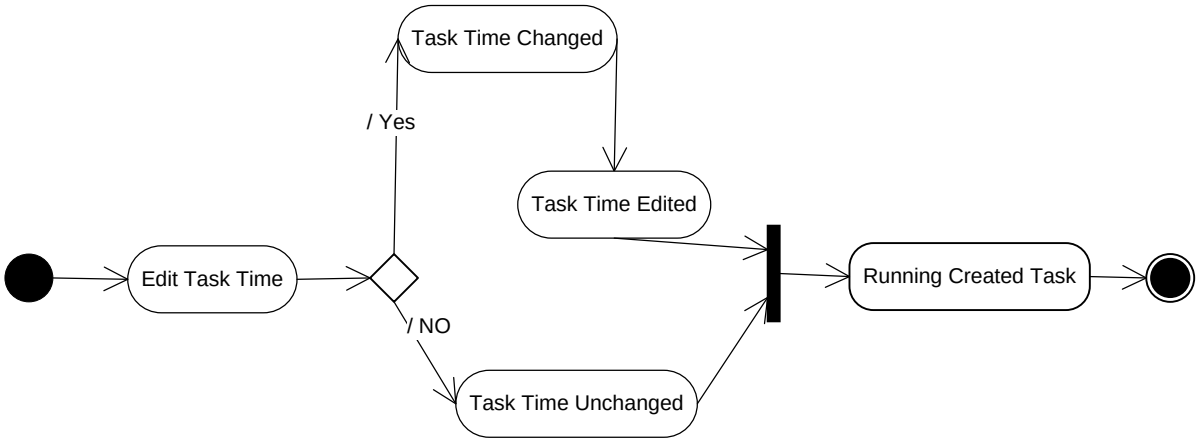
4.3 Activity Diagrams

4.3.1 New Task Activity Diagram



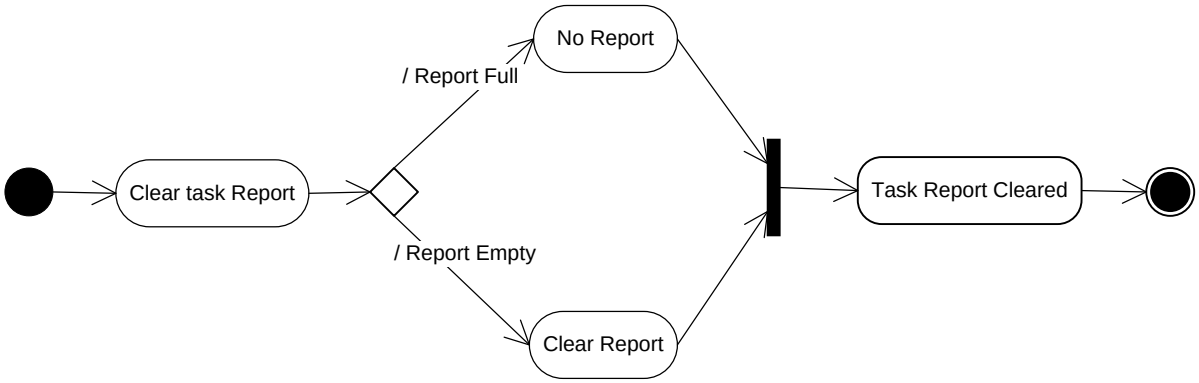
Activity Diagram Figure 1

4.3.2 Edit Time Activity Diagram



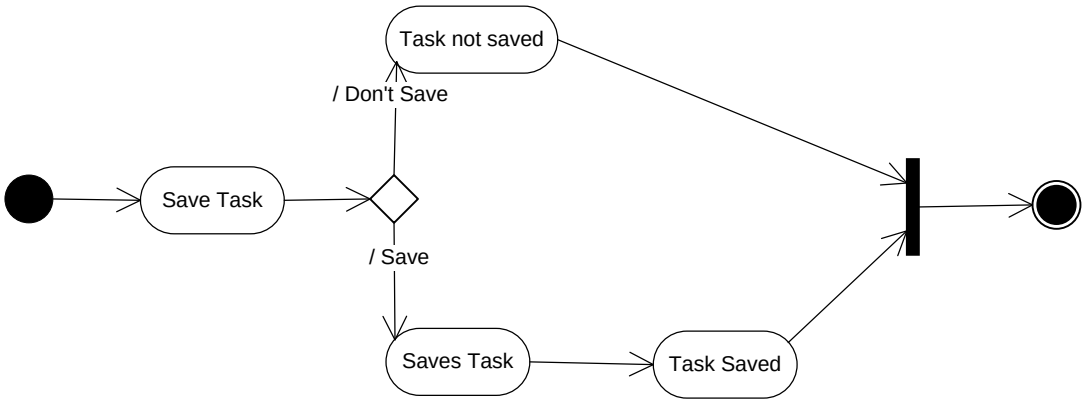
Activity Diagram Figure 2

4.3.3 Clear Task Activity Diagram



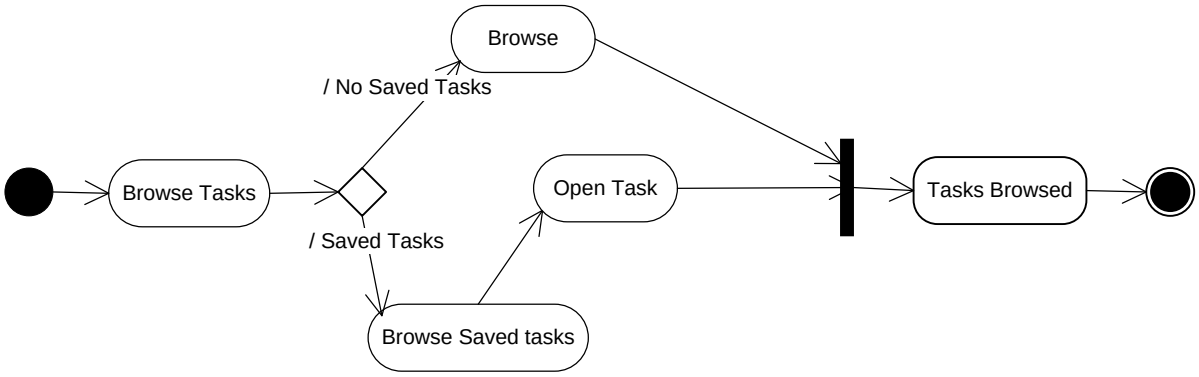
Activity Diagram Figure 3

4.3.4 Save As task Activity Diagram



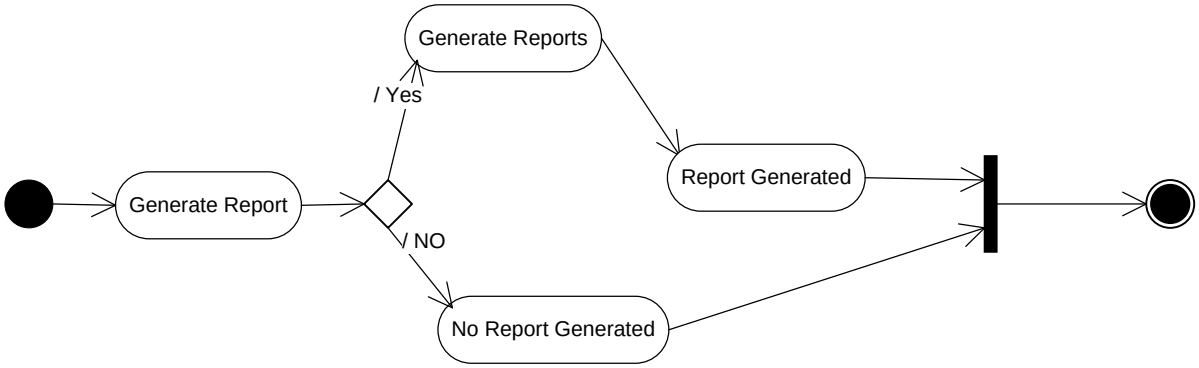
Activity Diagram Figure 4

4.3.5 Browser Activity Diagram



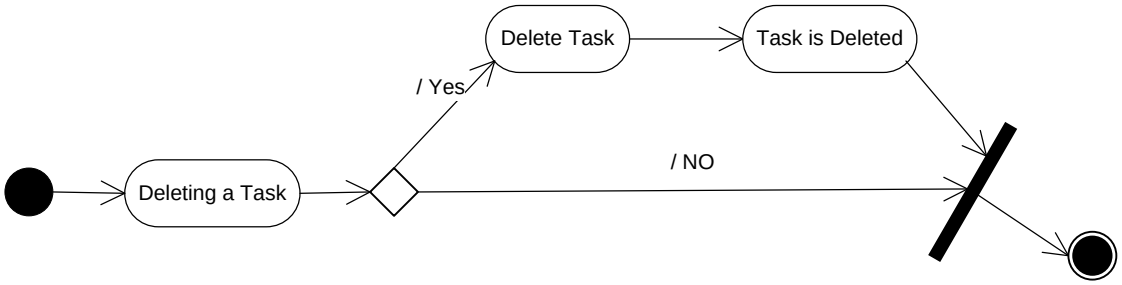
Activity Diagram Figure 5

4.3.6 Generate Report Activity Diagram



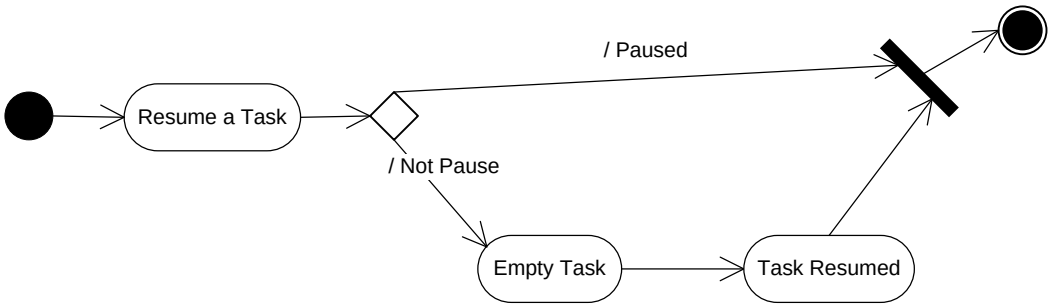
Activity Diagram Figure 6

4.3.7 DeleteTask Activity Diagram



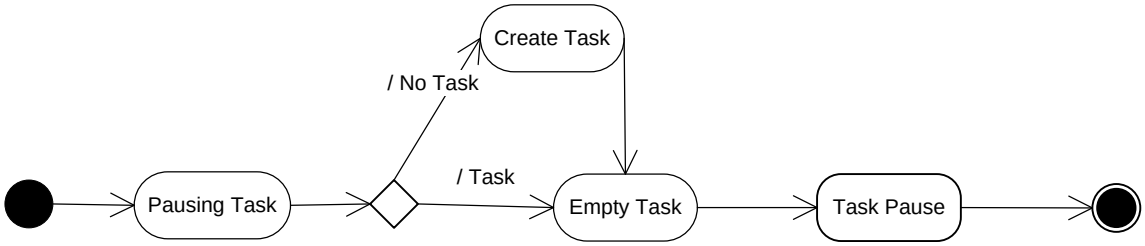
Activity Diagram Figure 7

4.3.8 Resume Task Activity Diagram



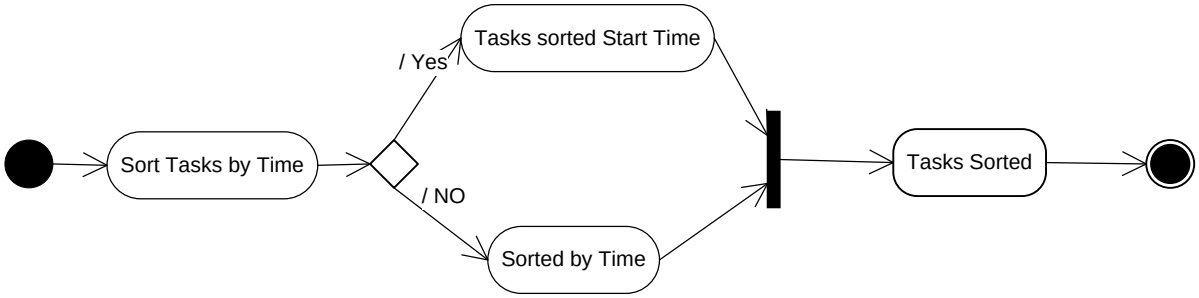
Activity Diagram Figure 8

4.3.9 Pause Task Activity Diagram



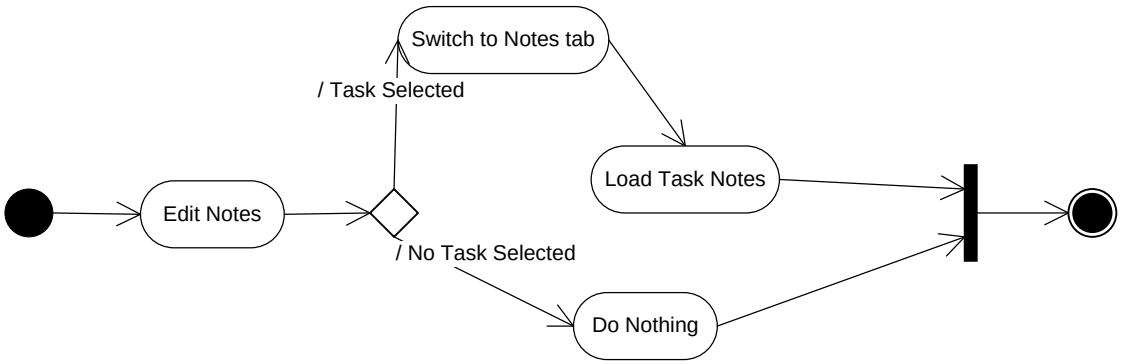
Activity Diagram Figure 9

4.3.10 Start Time Task Activity Diagram



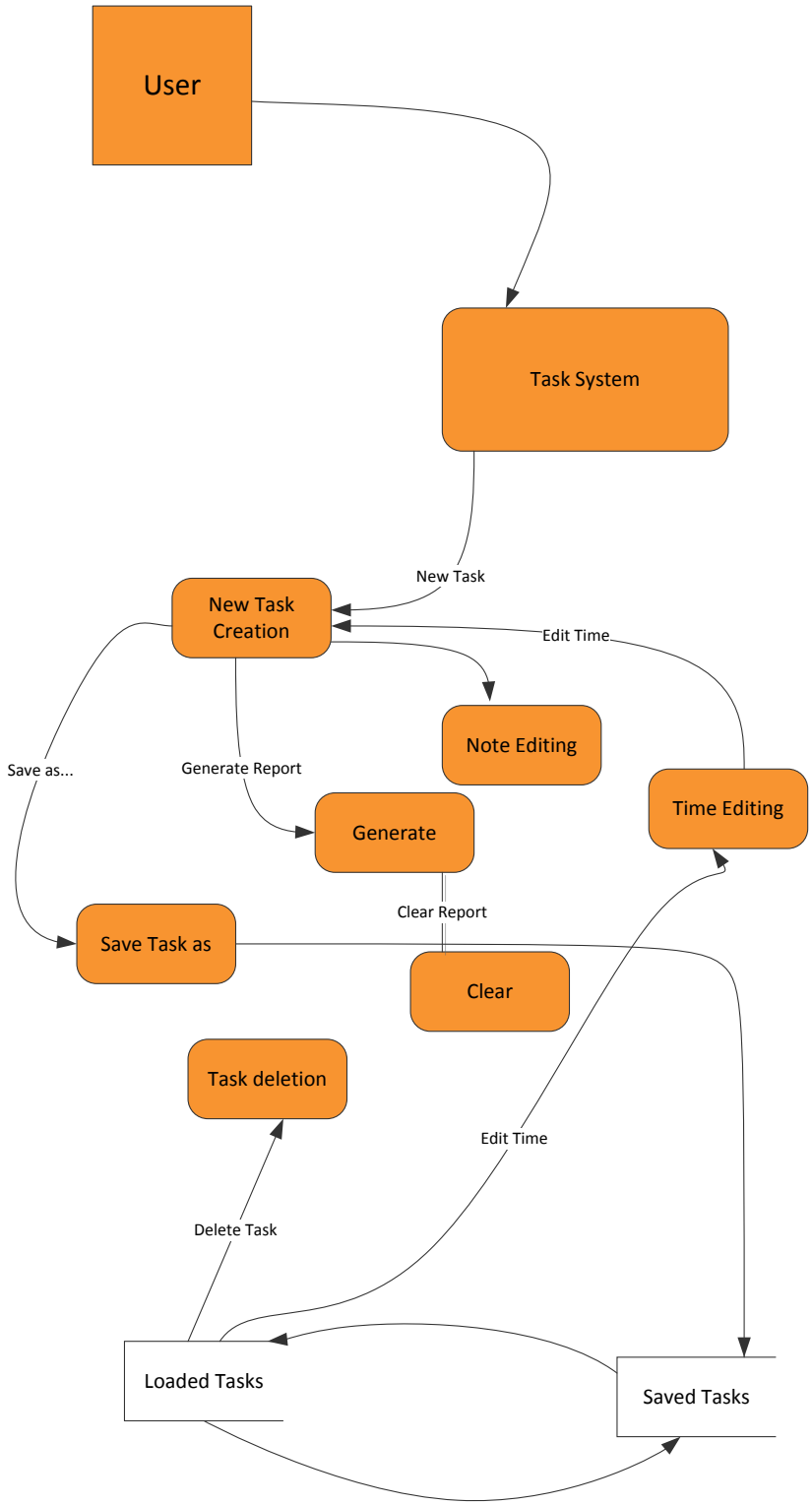
Activity Diagram Figure 20

4.3.11 Edit Notes Activity Diagram



Activity Diagram Figure 31

4.4 Data Flow Diagrams (DFD)



5. Change Management Process

Any changes to this document throughout the development process should be submitted to Mario for review and then to Jesse for editing, formatting and final addition to this document. When submitting changes to this document, please describe what needs to be added or changed, why and the section where the change needs to happen. All changes should be submitting to Mario before the end of day on the Tuesday prior to class.

If there are any misspellings, grammatical errors or other mistakes please follow the instructions above.

Before submitting this document all members of 0x00000001 must sign off on the final version.

A. Appendices

A.1 File Format Specification

File extension is .TSV, the standardized Tab Separated Value Format

01	Programming	04:16:28	+	0	Fixing bugs in product	1336813103573	1337570841096	15388689
02	Phonecall	00:02:15	-	1	Setting up meeting with client	1336888798573	1336888933573	135000
03	Presentation	00:15:43	-	2	Pitching product to client	1337062637573	1337063580573	943000
04	Meeting	00:38:21	-	3	Discussion with human resources	1337082516573	1337084817573	2301000
05	Q & A	00:02:15	-	4	Discussion with client	1337214417573	1337214417573	1210000
06	Backup	00:52:45	-	5	Copying files to external drive	1337215627573	1337218792573	3165000
07	Phonecall	00:00:11	-	6	Check in with boss	1337218795573	1337218806573	11000
08	Partitioning	00:28:34	-	7	Split drive into 3 sections	1337218807917	1337570289572	1714988
09	Reinstall	02:46:02	+	8	Installing OSes for the client	1337220522935	1337570292465	9962968
10	Setup software	01:59:10	+	9	Installing office suite and configuration	1337230485910	1337570297842	7150005
11	Stress test	32:15:48	+	10	Measuring reliability	1337449866934	1337570858225	116148936

Column 1: Row number (two digits)

Column 2: Task title

Column 3: Elapsed time (HH:MM:SS)

Column 4: Recent status (+ for in recent list, - otherwise). Last four recently accessed tasks.

Column 5: Unique task ID number

Column 6: Task notes (multi-line capable)

Column 7: Task start time in milliseconds from Epoch (January 1st, 1970 UTC)

Column 8: Task end time in milliseconds from Epoch (January 1st, 1970 UTC)

Column 9: Task total elapsed time in milliseconds

A.2 Eclipse Integration

The following appendix is from the Wiki hosted on Google Code.
These instructions pertain to setting up the development environment.

1. Install Java SE JDK

- Web browser->
- http://www.oracle.com/technetwork/java/javase/downloads/jdk-7u3-download-1501626.html
- Java Se Development Kit
- <platform (architecture)>
- Run Java installer

2. Install Eclipse "Indigo"

- Web browser->
- http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/indigosr2
- Download links
- <platform (architecture)>
- Run Eclipse installer

3. Install egit

- Eclipse->
  - Help->
    - Install new software->
      - Work with: *http://download.eclipse.org/egit/updates*
      - (Add)
        - Name: *egit*
        - **[OK]**
      - (✔) Eclipse Git Team Provider
      - **[Next]**
      - **[Next]**
      - (•) I accept the terms
      - **[Finish]**
      - **[Restart Now]**
- **4. Enable egit**

- Eclipse->
  - Window->
    - Customize perspective->
      - Command Groups Availability->
        - Git (✔)

    - Git Navigation Actions (✔)
- Menu Visibility->
    - Git (✔)
- Shortcuts->
    - Git (✔)
- Toolbar Visibility->
    - Git (✔)

5. Configure egit

- Eclipse->
- File->
- Import->
- (▻) Git->
- Projects from Git->
- **[Next]**
- URI->
- **[Next]**
- URI: *https://code.google.com/a/eclipselabs.org/p/timeme/*
- User: *myusername@gmail.com*
- Password: *MY-GENERATED-PASSWORD*
- (✔) Store in Secure Store
- **[Next]**
- (✔) master
- **[Next]**
- **[Browse]**
- Select folder
- **[Next]**
- (•) Input existing projects
- **[Next]**
- (✔) timeme
- (✔) Add project to working sets
- **[Select]**
- **[New]**
- Resource
- **[Next]**
- Working set name: *Time Me*
- **[Finish]**
- (✔) Time Me
- **[OK]**
- **[Finish]**

6. Install Mylyn Google Code connector

- Eclipse->
  - Help->
    - Install new software->
      - Work with: *http://knittig.de/googlecode-mylyn-connector/update/*
      - (Add)
        - Name: *gconnector*
        - **[OK]**
      - (✔) Nightly builds
      - **[Next]**
      - **[Next]**
      - (•) I accept the terms
      - **[Finish]**
      - Security Warning: **[OK]**
      - **[Restart Now]**

7. Configure Mylyn Google Code connector

- Eclipse->
  - Window->
    - Preferences->
      - (▻) Mylyn
        - Tasks
          - (✔) Synchronize with repositories every *5* minutes
          - Week Start: *Monday*
          - (✔) Highlight current line
      - **[Apply]**
      - **[OK]**
  - Window->
    - Open perspective->
      - Other...->
        - Team Synchronization
        - **[OK]**
  - Task Repositories (pane)->
    - ∇ (Menu)
      - Add Task Repository
        - Google Code
        - **[Next]**
        - Project URL: *https://code.google.com/a/eclipselabs.org/p/timeme/*
        - Label: *timeme*
        - ☐Anonymous
        - User ID: *myusername@gmail.com*
        - Password: *MY-GMAIL-PASSWORD*
        - (✔) Save Password
        - **[Validate Settings]** (Note: does not verify password)

        - **[Finish]**
    - Would you like to add a task?: **[Yes]**
        - (•) Use a predined query
            - Open issues
        - **[Finish]**
- Window->
    - Open perspective->
        - Other...->
            - Java (Default)
            - **[OK]**

A.3 Using SWT in Eclipse

The following appendix is from the Wiki hosted on Google Code.
These instructions pertain to setting up the development environment.

1. Download SWT library

- Web browser->
  - http://download.eclipse.org/eclipse/downloads/drops/R-3.7.2-201202080800/index.php#SWT
    - {**platform (architecture)**}
      - (http) ->
  - Download zip from {**mirror**} ->
  - Save to ~/Downloads
- **2. Download SWT library**

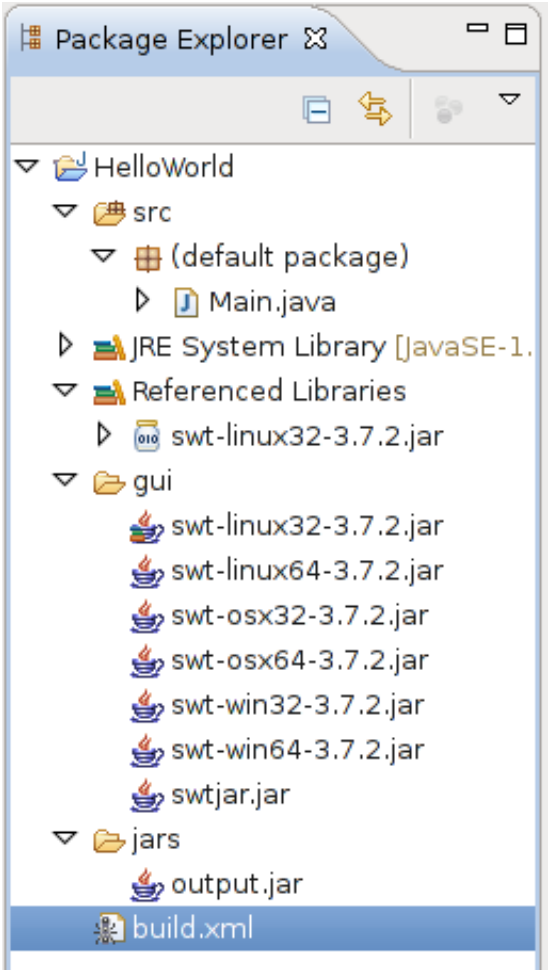
- File manager->
  - Right-click on **swt-**{**version**}**-**{**ui**}**-**{**platform**}**-**{**arch**}**.zip**
    - Unzip
  - Rename swt.jar to **swt-**{**platform**}{**arch**}**-**{**version**}**.jar**
  - Move jar to project folder Figure 1
- **3. Import SWT into Eclipse**

- Eclipse->
  - Package Explorer
    - Select project folder
      - Right-click->
        - Properties
          - Java Build Path->
            - Libraries->
              - Add JARs...->
                - Select: **swt-**{**platform**}{**arch**}**-**{**version**}**.jar**
                - **[OK]**
          - Java Compiler->
            - Compiler compliance level: **1.6**
          - **[OK]**

4. Compile .java source into .class files

- Eclipse->
  - Package Explorer
  - Select main source file
  - Run icon (or **[CTRL]**+**[F11]**)

Figure 1



Eclipse SWT directory structure

Figure 2

```
[user@host ~]$ find HelloWorld/ -type f
HelloWorld/build.xml
HelloWorld/gui/swt-win32-3.7.2.jar
HelloWorld/gui/swt-osx32-3.7.2.jar
HelloWorld/gui/swtjar.jar
HelloWorld/gui/swt-osx64-3.7.2.jar
HelloWorld/gui/swt-win64-3.7.2.jar
HelloWorld/gui/swt-linux32-3.7.2.jar
HelloWorld/gui/swt-linux64-3.7.2.jar
HelloWorld/.project
HelloWorld/jars/output.jar
HelloWorld/bin/Main$1.class
HelloWorld/bin/Main.class
HelloWorld/src/Main.java
HelloWorld/.classpath
HelloWorld/.settings/org.eclipse.jdt.core.prefs
```

Full SWT directory structure

Cross-platform packaging

1. Download SWT library for all platforms

- Repeat steps 1-2 in Using SWT in Eclipse for all platforms.
- List should include:
- swt-linux32-{version}.jar
- swt-linux64-{version}.jar
- swt-osx32-{version}.jar
- swt-osx64-{version}.jar
- swt-win32-{version}.jar
- swt-win64-{version}.jar
  NOTE: these filenames must match exactly

2. Download swtjar.jar

- Web browser->
  - http://mchr3k.github.com/swtjar/
    - Download swtjar.jar
- Save to project folder Figure 1

3. Create ANT build file

- Eclipse->
  - Package Explorer
    - Select project folder
      - Right-click->
    - New->
      - File->
      - Filename: **build.xml**
    - **[OK]**

NOTE: Uses file structure from Figure 2
build.xml contents:

```
<?xml version="1.0"?>
<project name="HelloWorld">

<taskdef name="swtjar" classname="org.swtjar.ant.SWTJarTask" classpath="./gui/swtjar.jar"/>

        <!-- Package cross platform SWT Jar -->
        <swtjar jarfile="./jars/output.jar" targetmainclass="Main" swtversion="3.7.2">

                  <!-- Application Classes -->
                  <fileset dir="./bin/" includes="**/*.class" />

                  <!-- SWT Jars -->
                  <fileset dir="./gui" includes="swt-*-3.7.2.jar" />

        </swtjar>

</project>
```

4. Compile .java source into .class files

- Eclipse->
  - Package Explorer
    - Select main .java source file
  - Run icon (or **[CTRL]+[F11]**)

5. Build "portable" jar package

- Eclipse->
  - Package Explorer
    - Select build.xml file
  - Run icon (or **[CTRL]+[F11]**)
- Distribute output.jar

A.4 Code References

Description of containers here:
http://java.sun.com/products/jfc/tsc/articles/containers/#swing_containers

"Repurposed" the example code:
http://docs.oracle.com/javase/tutorial/uiswing/examples/layout/BoxAlignmentDemoProject/src/layout/BoxAlignmentDemo.java

BoxLayout:
http://docs.oracle.com/javase/tutorial/uiswing/layout/box.html

"in-line" HTML trick to make both the tabs and buttons bigger:
http://weblogs.java.net/blog/xuanyun/archive/2009/01/change_the_tab_1.html

button event handler:
http://www.codebeach.com/2009/03/introduction-to-java-swing.html

tab event handler:
http://www.exampledepot.com/egs/javax.swing/tabbed_TpEvt.html

list event handler:
http://www.devdaily.com/java/java-jlist-listselectionlistener-event

table event handler:
http://www.java2s.com/Tutorial/Java/0240__Swing/TableSelectionEventsandListeners.htm

Read file (line by line):
http://www.roseindia.net/java/beginners/java-read-file-line-by-line.shtml

Write file:
http://www.daniweb.com/software-development/java/code/217078

Append file:
http://www.mkyong.com/java/how-to-append-content-to-file-in-java/

Check if file exists:
http://stackoverflow.com/questions/1816673/how-do-i-check-if-a-file-exists-java-on-windows

Set JFileChooser directory:
http://docs.oracle.com/javase/6/docs/api/javax/swing/JFileChooser.html#setCurrentDirectory%28%29

Current working directory:
http://www.exampledepot.com/egs/java.io/CurDir.html

Single instances:
http://www.java-forums.org/awt-swing/4407-jfilechooser-remember-location.html

String to File:
http://www.coderanch.com/t/383573/java/java/convert-String-File-object

Working timer:
http://stackoverflow.com/questions/2576353/stop-a-stopwatch/2576909#2576909

Standard conventions:
http://www.mc.vanderbilt.edu/infocntr/infointgr/AppDevelopment/javaCodingStd.html#secName

SWT select all:
http://stackoverflow.com/questions/4143751/how-to-restore-default-keybindings-ctrla-ctrlc-etc-for-widgets-in-swt

SWT background thread:
http://book.javanb.com/swt-the-standard-widget-toolkit/ch05lev1sec7.html

SWT xygraph:
http://swt-xy-graph.googlecode.com/git/PureJava/org.csstudio.swt.xygraph/html/GettingStarted.html

SWT column sort:
http://www.java-forums.org/swt/10050-sorting-swt-table-column.html

Close gracefully:
http://stackoverflow.com/questions/483173/how-can-i-get-my-basic-swt-application-to-exit-properly-in-mac-os-x-10-5-6

SWT filechooser:
http://stackoverflow.com/questions/6872141/selecting-multiple-files-in-filechooser

SWT edit table:
http://git.eclipse.org/c/platform/eclipse.platform.swt.git/tree/examples/org.eclipse.swt.snippets/src/org/eclipse/swt/snippets/Snippet88.java

string regex:
http://www.deitel.com/articles/java_tutorials/20060218/index.html

calendar date:
http://www.mkyong.com/java/java-how-to-get-current-date-time-date-and-calender/

icon fix:
http://stackoverflow.com/questions/4521973/how-do-i-add-an-icon-as-a-classpath-resource-to-an-swt-window-created-with-windo

sort 2D array:
http://stackoverflow.com/questions/4907683/sort-a-two-dimensional-array-based-on-one-column

newline:
http://leepoint.net/notes-java/io/10file/sys-indep-newline.html

confirm dialog:
http://www.vogella.com/articles/EclipseDialogs/article.html#dialogs_swt

A.5 Scripts

A.5.1 gitcommits-pdf

```
#!/bin/bash
########################
### gitcommits-pdf ####
########################
### Team 0x00000001 ###
########################

shared="/usr/share/gitbook";
path="/dev/shm/commits";

if [ -z  "$1" ];
then echo "USAGE: $(basename $0) [template] [title] {[NONE]|[PATH]}";
fi

pathDir=$(dirname "$path");
if [ ! -d "$pathDir" ];
then mkdir -p "$pathDir";
fi

if [ -d "$3" ];
then cd "$3";
fi

git --no-pager log --pretty="format:[START
commit][author=%an][time=%at][message=%s][hash=%H]" --shortstat >
"${path}.txt";

oldPWD="$PWD";

if [ -d "$shared" ];
then cd "$shared";
else cd "$pathDir";
fi

if [ -d "$1" ];
then template="$1";
else
  curl "http://timeme.eclipselabs.org.codespot.com/git/scripts/github.tar.gz"
| tar xz;
  template="github";
fi

if [ ! -z "$2" ];
then title="$2";
fi

if [ -z "$template" ] || [ -z "$title" ];
then echo "Error"; exit 1;
fi

python2 "gitbook.py" "$template" "$title" "${path}.txt" "${path}.html";
wkhtmltopdf -s Letter -b "${path}.html" "${path}.pdf";
cd "$oldPWD";
evince "${path}.pdf";

### END ###
```

A.5.2 gitstats-pdf

```
#!/bin/bash
########################
### gitstats-pdf ######
########################
version="0.2" #########
### Team 0x00000001 ###
########################

###############
## Config
inputDir="$HOME/project";
tmpDir="/dev/shm/stats";
outputDir="/dev/shm/";

## Valid colors are
#color="black"
color="blue"
#color="green"
#color="purple"
#color="red"
#color="pink"
#color="orange"
#color="yellow"
#color="white"

###############
## Functions
exit_script()
{
  echo "$1 failed, exiting"; exit 1;
}

check_depends()
{
  for depend in $depends;
  do
    ifcheck=$(which $depend 2>/dev/null);
    if [ -z "$ifcheck" ] && [ "$depend" = "convert" ];
    then echo "Missing dependency: imagemagick not installed"; missing="true";
    else
      if [ -z "$ifcheck" ];
      then echo "Missing dependency: $depend not installed"; missing="true";
      fi;
    fi;
  done;
  if [ "$missing" = "true" ];
  then echo "FATAL ERROR: install missing package(s)"; exit 1;
  fi;
}

color_map()
{
  if [ -z "$color" ];
  then exit_script "Color map";
  else
    if [ "$color" = "black" ];
    then red="0"; green="0"; blue="0";
  else
    if [ "$color" = "blue" ];
    then red="0"; green="0"; blue="255";
  else
```

```
    if [ "$color" = "green" ];
    then red="0"; green="255"; blue="0";
  else
    if [ "$color" = "purple" ];
    then red="127"; green="0"; blue="255";
  else
    if [ "$color" = "red" ];
    then red="127"; green="0"; blue="255";
  else
    if [ "$color" = "pink" ];
    then red="127"; green="0"; blue="255";
  else
    if [ "$color" = "orange" ];
    then red="127"; green="0"; blue="255";
  else
    if [ "$color" = "yellow" ];
    then red="127"; green="0"; blue="255";
  else
    if [ "$color" = "white" ];
    then red="127"; green="0"; blue="255";
    fi; fi; fi; fi; fi; fi; fi; fi; fi;
  fi;
}


###############
## Check for dependencies
depends="convert evince gitstats sed wkhtmltopdf";
check_depends;
color_map;

## Sanity check
if [ ! -d "$inputDir" ];
then echo "$inputDir does not exist"; exit 1;
fi

if [ ! -d "$tmpDir" ];
then mkdir "$tmpDir"; [ -d "$tmpDir" ] || exit_script "Sanity check";
fi

if [ ! -d "$outputDir" ];
then mkdir "$outputDir"; [ -d "$outputDir" ] || exit_script "Sanity check";
fi

###############
## Generate statistics
cd "$inputDir";
echo "##############################";
echo "### Generating statistics ###";
echo "##############################";
gitstats . "$tmpDir" || exit_script "gitstats";
echo -e "[DONE]\n\n";
cd "$tmpDir";

## Recolor tables
echo "#########################";
echo "### Recoloring tables ###";
echo "#########################";
#sed -i "s/rgb([[:digit:]]*, /rgb($red, /g" "$tmpDir/activity.html" ||
exit_script "sed";
sed -i "s/rgb([[:digit:]]/rgb(/g" "$tmpDir/activity.html" || exit_script
"sed";
sed -i "s/, [[:digit:]]*, [[:digit:]]*)/, $green, $blue)/g"
"$tmpDir/activity.html" || exit_script "sed";
```

```
sed -i "s/background-color: red/background-color: $color/g"
"$tmpDir/activity.html" || exit_script "sed";
echo -e "[DONE]\n\n";

## Recolor graphs
echo "##########################";
echo "### Recoloring graphs ###";
echo "##########################";
IFS=$(echo -en "\n\b");
for i in $(ls "$tmpDir"/*.png);
do
  convert -fill $color -opaque red "$i" "${i}_" && mv "${i}_" "$i" ||
exit_script "imagemagick";
done;
echo -e "[DONE]\n\n";

## Convert to PDF
echo "##########################";
echo "### Converting to PDF ###";
echo "##########################";
p1="$tmpDir/index.html";
p2="$tmpDir/activity.html";
p3="$tmpDir/authors.html";
p4="$tmpDir/files.html";
p5="$tmpDir/lines.html";
p6="$tmpDir/tags.html";

## Sanity check
if [ ! -f "$p1" ] || [ ! -f "$p2" ] || [ ! -f "$p3" ] || [ ! -f "$p4" ] || [ !
-f "$p5" ] || [ ! -f "$p6" ];
then echo "Missing file(s) in $tmpDir"; exit 1;
fi

wkhtmltopdf -s Letter -b "$p1" "$p2" "$p3" "$p4" "$p5" "$p6"
"$outputDir/output.pdf" || exit_script "wkhtmltopdf";
echo -e "[DONE]\n\n";

## Open PDF for printing
evince "$outputDir/output.pdf" > /dev/null 2>&1 &

### END ###
```

A.6 GIT Statistics

A.6.1 Summary

Project name:
timeme
Generated:
2012-05-22 06:43:36 (in 95 seconds)
Generator:
GitStats (version ad7efbb), git version 1.7.6.4, gnuplot 4.4 patchlevel 3
Report Period:
2012-03-01 01:51:02 to 2012-05-16 13:22:30
Age:
77 days, 41 active days (53.25%)
Total Files:
461
Total Lines of Code:
142413 (159324 added, 16911 removed)
Total Commits:
154 (average 3.8 commits per active day, 2.0 per all days)
Authors:
3 (average 51.3 commits per author)

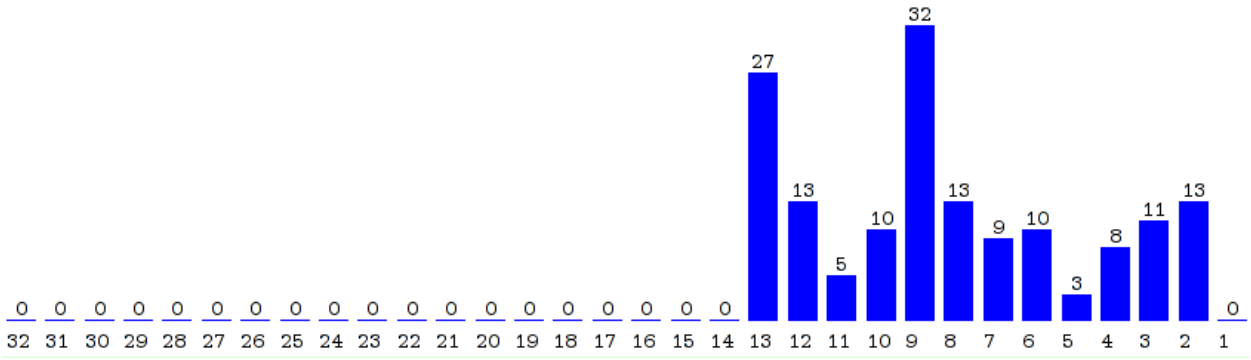
A.6.2 Lines

Ohloh Line Count Summary

Language	Files	Code	Comment	Comment %	Blank	Total
java	12	1478	164	10.0%	239	1881
Total	12	1478	164	10.0%	239	1881

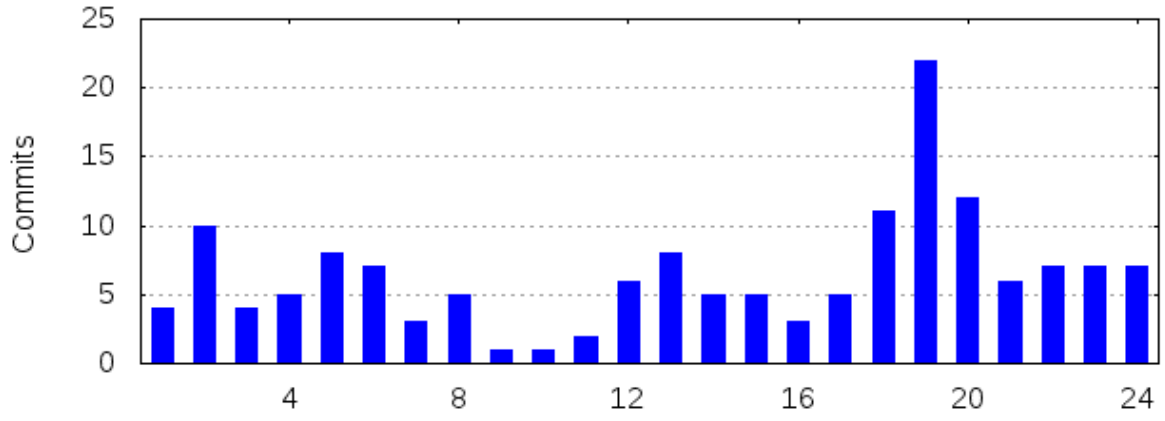
A.6.3 Activity

Weekly Activity

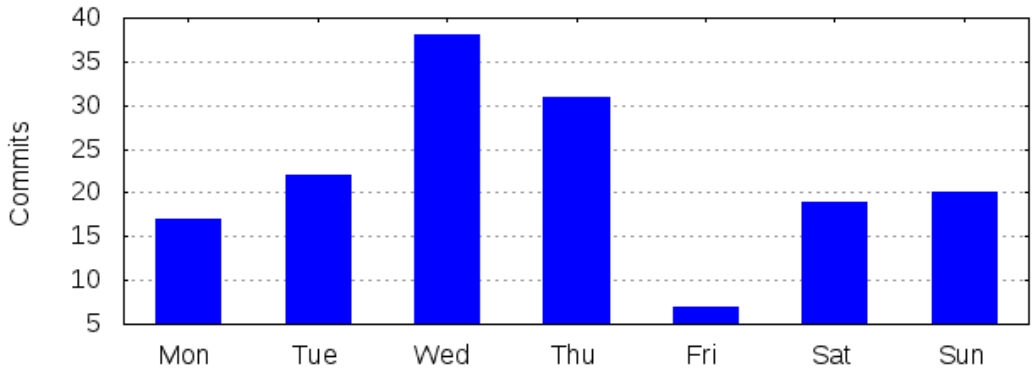


Hour of Day

Hour	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Commits	4	10	4	5	8	7	3	5	1	1	2	6	8	5	5	3	5	11	22	12	6	7	7	7
%	2.60	6.49	2.60	3.25	5.19	4.55	1.95	3.25	0.65	0.65	1.30	3.90	5.19	3.25	3.25	1.95	3.25	7.14	14.29	7.79	3.90	4.55	4.55	4.55



Day of Week

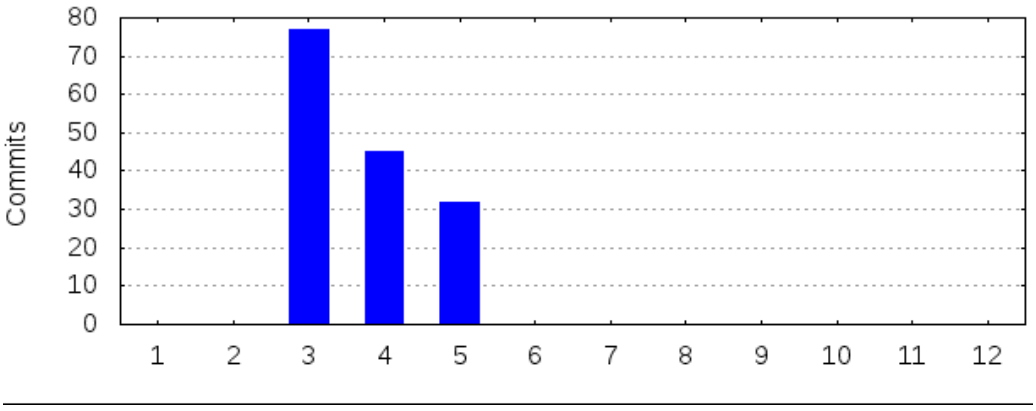


Day	Total (%)
Mon	17 (11.04%)
Tue	22 (14.29%)
Wed	38 (24.68%)
Thu	31 (20.13%)
Fri	7 (4.55%)
Sat	19 (12.34%)
Sun	20 (12.99%)

Hour of Week

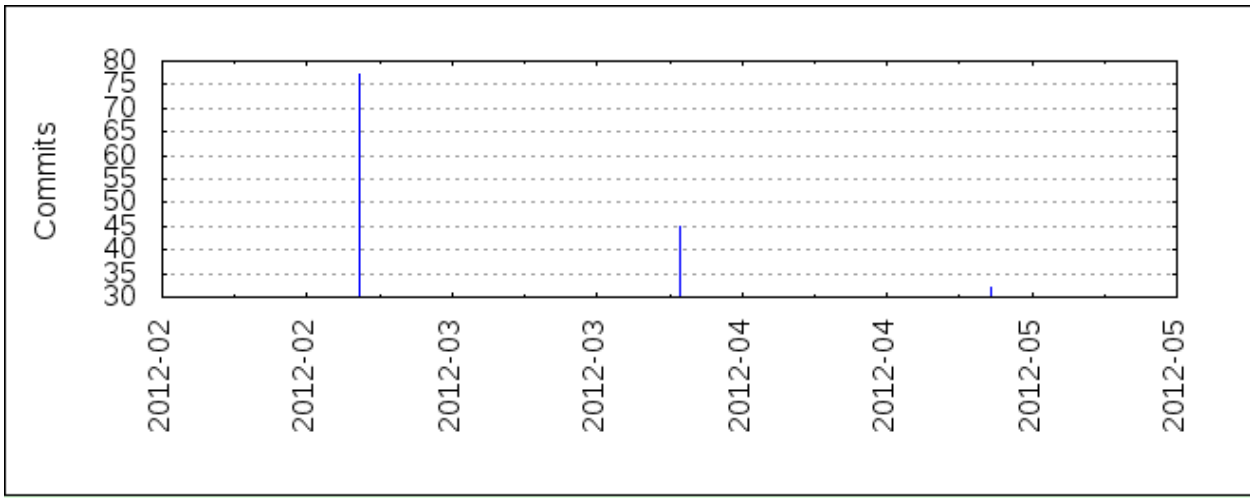
Weekday	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Mon	2				5	3											1	1			1		2	2
Tue		1	3		1	2						1		1	2	1	3	2	1	1				3
Wed	2					1	2		1	1			8	2	2			4	8	4	2	1		
Thu		2		1			1	5			1	5		1	1	2			7	2		1	2	
Fri		1		2	2																1	1		
Sat				2										1				4	5	1		3	3	
Sun		6	1			1					1						1		1	4	2	1		2

Month of Year



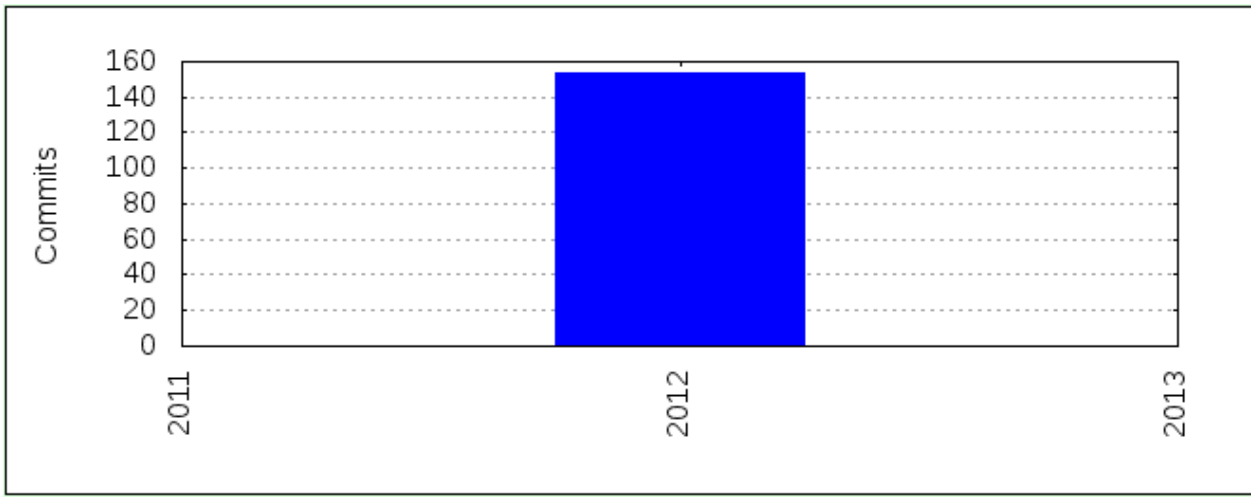
Month	Commits (%)
1	0 (0.00 %)
2	0 (0.00 %)
3	77 (50.00 %)
4	45 (29.22 %)
5	32 (20.78 %)
6	0 (0.00 %)
7	0 (0.00 %)
8	0 (0.00 %)
9	0 (0.00 %)
10	0 (0.00 %)
11	0 (0.00 %)
12	0 (0.00 %)

Commits by month



Month	Commits	Lines added	Lines removed
2012-05	32	130174	5034
2012-04	45	23367	9181
2012-03	77	5783	2696

Commits by year



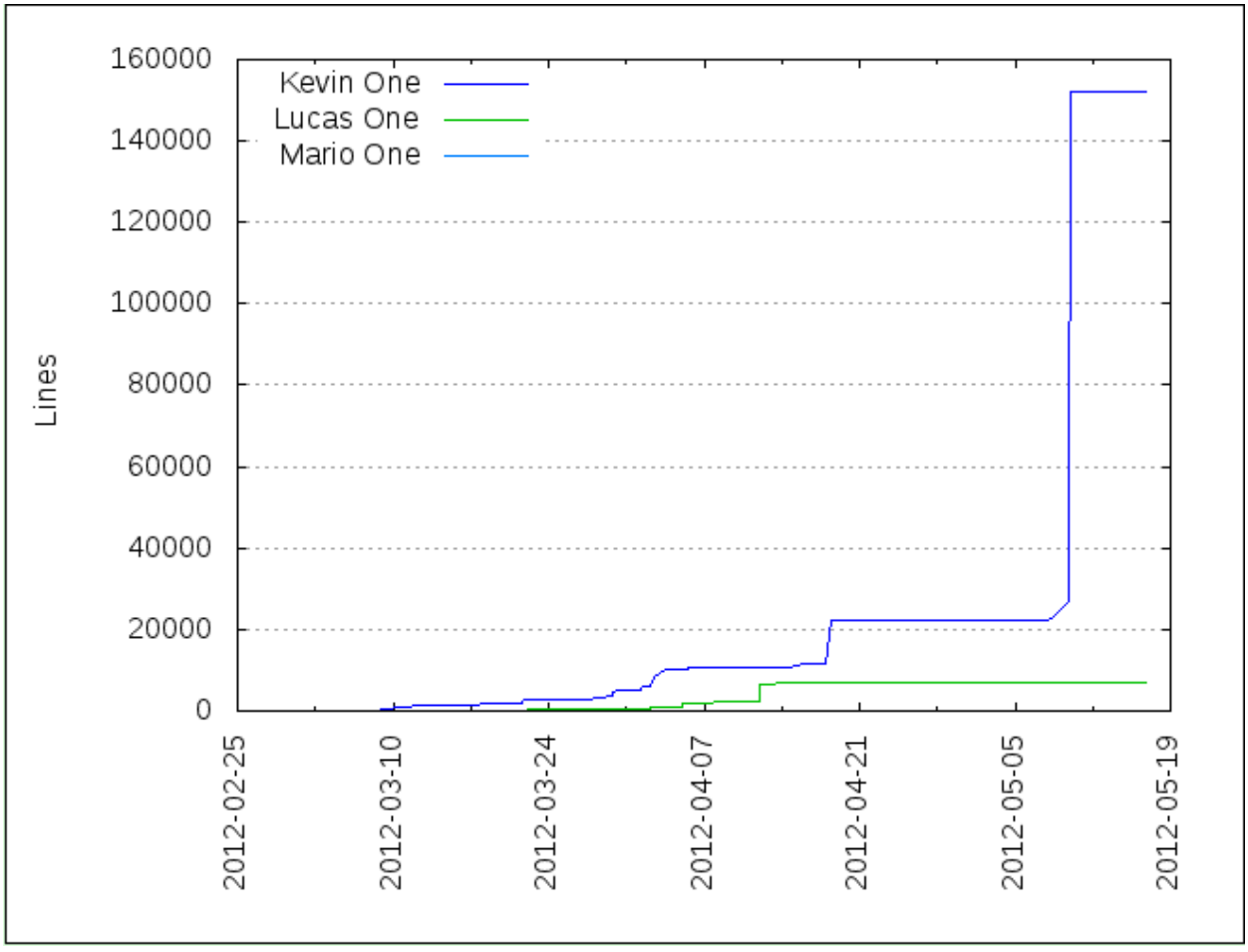
Year	Commits (% of all)	Lines added	Lines removed
2012	154 (100.00%)	159324	16911

A.6.4 Authors

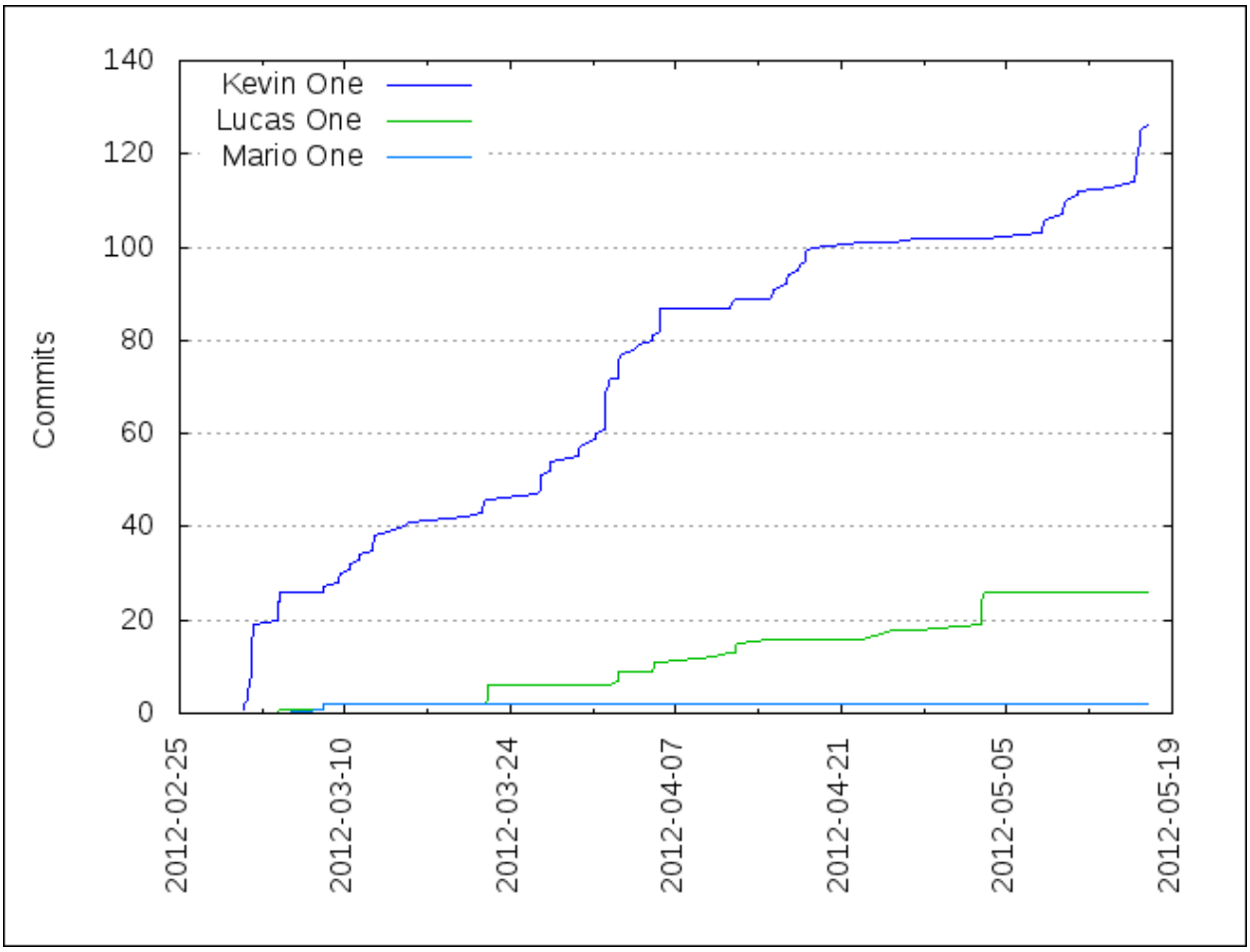
List of Authors

Author	Commits (%)	+ lines	- lines	First commit	Last commit	Age	Active days	# by commits
Kevin One	126 (81.82%)	152344	10927	2012-03-01	2012-05-16	76 days, 11:31:28	37	1
Lucas One	26 (16.88%)	6978	5982	2012-03-04	2012-05-02	59 days, 17:23:25	11	2
Mario One	2 (1.30%)	2	2	2012-03-07	2012-03-07	0:01:37	1	3

Cumulated Added Lines of Code per Author



Commits per Author



Author of Month

Month	Author	Commits (%)	Next top 5	Number of authors
2012-05	Kevin One	24 (75.00% of 32)	Lucas One	2
2012-04	Kevin One	33 (73.33% of 45)	Lucas One	2
2012-03	Kevin One	69 (89.61% of 77)	Lucas One, Mario One	3

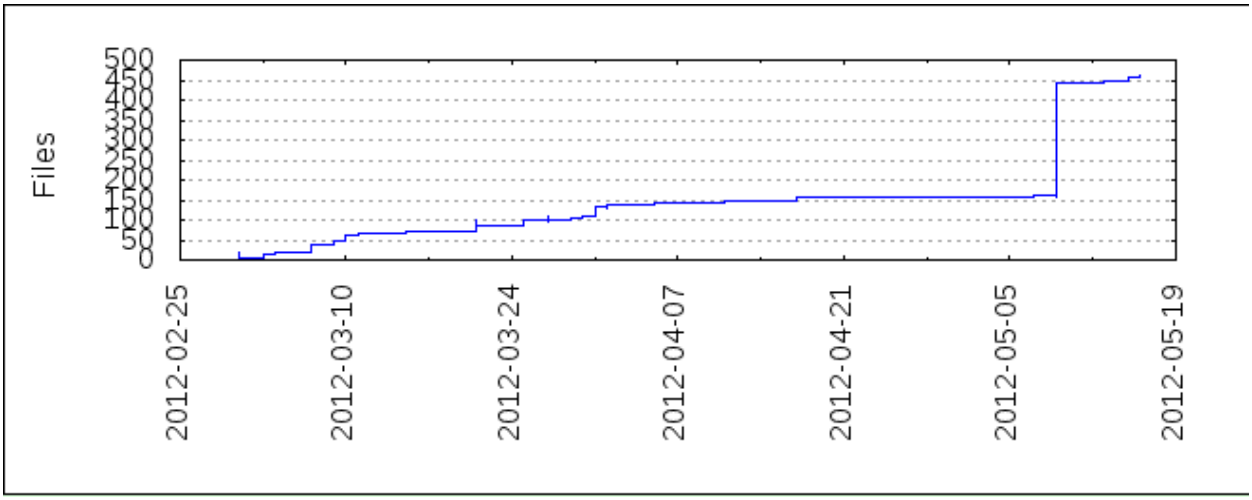
Author of Year

Year	Author	Commits (%)	Next top 5	Number of authors
2012	Kevin One	126 (81.82% of 154)	Lucas One, Mario One	3

A.3.5 Files

Total files:
461
Total lines:
142413
Average file size:
940167.98 bytes

File count by date



Extensions

Extension	Files (%)	Lines (%)	Lines/file
	8 (1.74%)	174 (0.12%)	21
PDF	1 (0.22%)	10210 (7.17%)	10210
apng	1 (0.22%)	63 (0.04%)	63
class	8 (1.74%)	384 (0.27%)	48
conf	1 (0.22%)	1 (0.00%)	1
css	1 (0.22%)	474 (0.33%)	474
ep	1 (0.22%)	180 (0.13%)	180
gif	4 (0.87%)	62 (0.04%)	15
html	62 (13.45%)	11703 (8.22%)	188
jar	56 (12.15%)	1493372 (1048.62%)	26667
java	48 (10.41%)	6467 (4.54%)	134
mgc	1 (0.22%)	129 (0.09%)	129
mgi	1 (0.22%)	38 (0.03%)	38
pdf	22 (4.77%)	261934 (183.93%)	11906
png	195 (42.30%)	42392 (29.77%)	217
prefs	1 (0.22%)	4 (0.00%)	4
psd	1 (0.22%)	14 (0.01%)	14
svg	5 (1.08%)	0 (0.00%)	0
tiff	1 (0.22%)	0 (0.00%)	0
tsv	2 (0.43%)	10 (0.01%)	5
xcf	34 (7.38%)	55512 (38.98%)	1632
xml	6 (1.30%)	17 (0.01%)	2
zip	1 (0.22%)	10703 (7.52%)	10703

A.7 GIT Commit Log

TimeMe

2012-05-16

TimeMe 1.0 * TimeMe_1.0.jar * Update compiled class files * Cleanup unused code **Kevin One** - 01:22:30 PM	Files Modified 10 Lines Added 0 Lines Removed 4
Zero19.jar * Fix for running status when editing title/notes * Added screenshots to wiki **Kevin One** - 12:42:06 AM	Files Modified 6 Lines Added 5 Lines Removed 0

2012-05-15

Zero18.jar * Last commit was actually Zero17.jar * Stripped out debug functionality **Kevin One** - 11:45:26 PM	Files Modified 6 Lines Added 4 Lines Removed 28
Zero18.jar * Minor GUI tweak to accommodate larger themes **Kevin One** - 11:16:25 PM	Files Modified 2 Lines Added 5 Lines Removed 5
Zero16.jar * TimeMe 1.0 RC7 * Changed edit notes to add task to recent list **Kevin One** - 11:00:27 PM	Files Modified 2 Lines Added 18 Lines Removed 9
Zero15.jar TimeMe 1.0 RC6 * Update current task before generating report **Kevin One** - 07:36:00 PM	Files Modified 2 Lines Added 6 Lines Removed 0
Zero14.jar (again) * TimeMe 1.0 RC5-2 * Fix for delete task crash when no task selected **Kevin One** - 06:05:32 PM	Files Modified 2 Lines Added 18 Lines Removed 15
Zero14.jar * TimeMe 1.0 RC5 * Fix for clock of paused recent tasks **Kevin One** - 05:52:08 PM	Files Modified 2 Lines Added 1 Lines Removed 1
* Minor fix to select most recent task when current task deleted **Kevin One** - 04:59:03 PM	Files Modified 1 Lines Added 1 Lines Removed 0
Zero13.jar * TimeMe 1.0 RC4 * Fixed edited current task not saved * Fixed paused recent task only 6 digits * Fixed status is running while paused * Fixed stop timer if current task deleted * Changed pauseResume to Start on initial task * Added save title and notes in real time **Kevin One** - 04:42:19 PM	Files Modified 6 Lines Added 48 Lines Removed 19
Zero12.jar * TimeMe 1.0 RC3 * New tasks as Untitled-# * Fix for empty recent task crash * Added status to running task * Thanks to all the beta testers (: **Kevin One** - 02:54:43 PM	Files Modified 7 Lines Added 57 Lines Removed 54
Zero11.jar * TimeMe 1.0 RC2 * Added empty table sanity check (crash fix) * Changed row# to start at 01 instead of 00 * Minor cleanup **Kevin One** - 01:08:12 PM	Files Modified 6 Lines Added 27 Lines Removed 26
Zero10.jar * TimeMe 1.0 Release Candidate 1 * Added current task to recent list and table * Added update current task feature * Improved load from file support * Modified new task behavior * Code cleanup * etc. **Kevin One** - 11:30:48 AM	Files Modified 10 Lines Added 119 Lines Removed 53

2012-05-13

Zero9.jar * Buggy fix for recent tasks jumping around * Some work on dealing with initial task and new tasks * Renamed timeElapsed to total for uniformity * Code cleanup * Some other minor changes and testing * Unfinished. **Kevin One** - 04:00:05 PM	Files Modified 9 Lines Added 76 Lines Removed 57

2012-05-10

Zero8.jar (again) * Added fix for clock with recent tasks when stopwatch paused **Kevin One** - 03:24:30 PM	Files Modified 2 Lines Added 1 Lines Removed 0
Zero8.jar * Fix for crash for recent tasks after config loaded * Removed inaccurate table auto-select * Code cleanup **Kevin One** - 03:07:35 PM	Files Modified 5 Lines Added 9 Lines Removed 9

2012-05-09

Diagrams, Diagrams, Diagrams * Sorted diagrams * Added state diagrams **Kevin One** - 06:06:43 PM	Files Modified 361 Lines Added 112695 Lines Removed 0
* javadoc HTML * uml sequence diagrams **Kevin One** - 12:13:25 PM	Files Modified 220 Lines Added 12179 Lines Removed 0
* Some diagrams **Kevin One** - 12:10:17 PM	Files Modified 10 Lines Added 167 Lines Removed 0
* Minor code cleanup * Deprecated file cleanup **Kevin One** - 12:03:10 PM	Files Modified 82 Lines Added 4172 Lines Removed 4174

2012-05-07

Zero7.jar * Added features to closeDialog * Fixed tsv newline (Windows) * Moved browse and save dialog around * Removed unload feature (again) * Removed autosave-on-new-task feature * Cleanup **Kevin One** - 10:27:19 PM	Files Modified 6 Lines Added 137 Lines Removed 132
Zero6.jar * Added auto-save function * Added save confirm-on-close dialog * Renamed taskList to recentTasks * More cleanup **Kevin One** - 08:50:01 PM	Files Modified 9 Lines Added 126 Lines Removed 76
Zero5.jar * Fix for start times * Cleanup deprecated code **Kevin One** - 05:27:59 PM	Files Modified 5 Lines Added 113 Lines Removed 241
* Zero4.jar * Fixed report output in Windows * Removed write support from LoadFile * Included Lucas' Zero3.jar **Kevin One** - 04:50:01 PM	Files Modified 4 Lines Added 45 Lines Removed 53

2012-05-02

Misses a line in the restructure **Lucas One** - 07:16:30 PM	Files Modified 1 Lines Added 1 Lines Removed 0
minor change to start clock ticking when new task is pushed **Lucas One** - 06:00:49 PM	Files Modified 1 Lines Added 2 Lines Removed 0
integrated TaskObject.createTask into TaskObject.saveTask to remove a bug and to improve structure. **Lucas One** - 02:12:53 PM	Files Modified 2 Lines Added 31 Lines Removed 35
Added to the SetTime listener so that it updated the actual elapsed time as well as the displayed value. **Lucas One** - 01:36:56 PM	Files Modified 2 Lines Added 17 Lines Removed 2
Changed addRecent function to remove duplicates from the recent task list **Lucas One** - 12:37:25 PM	Files Modified 1 Lines Added 6 Lines Removed 2
Added proper task saving behavior to the Edit Notes button **Lucas One** - 12:26:42 PM	Files Modified 1 Lines Added 10 Lines Removed 4
Removing unintentional additions to the branch. **Lucas One** - 12:15:06 PM	Files Modified 3 Lines Added 1 Lines Removed 3
Ironed out the task switching issues in the recent task Pane. Created methods in TaskObject to handle each step to make the code more readable and object oriented. **Lucas One** - 12:06:19 PM	Files Modified 5 Lines Added 77 Lines Removed 32

2012-04-27

Zero2.jar * Fix for high CPU usage in background thread **Kevin One** - 03:16:26 AM	Files Modified 2 Lines Added 9 Lines Removed 4

2012-04-25

ugly workaround to keep the values returned by getSelectedIndex within the values of that table.

Lucas One - 12:06:29 AM

Files Modified	1
Lines Added	3
Lines Removed	1

2012-04-23

Clicks on the recent list now check to see if the current task exists and saves it to the right place before loading the new task.

Lucas One - 11:52:41 PM

Files Modified	2
Lines Added	67
Lines Removed	4

2012-04-22

Zerol.jar * Added sort function to report generation * Disabled debug in report textbox * Modified report textbox behavior

Kevin One - 10:54:19 AM

Files Modified	7
Lines Added	76
Lines Removed	13

2012-04-18

Added git-commit and git-stats PDFs

Kevin One - 05:27:16 AM

Files Modified	2
Lines Added	10679
Lines Removed	0

2012-04-17

MegaX9 - untested. **Kevin One** - 05:16:09 PM	Files Modified 1 Lines Added 0 Lines Removed 0
* Added test support for titlebar clock (warning: may be unstable) **Kevin One** - 04:53:03 PM	Files Modified 2 Lines Added 23 Lines Removed 6
Quick fix for collapse/expand bug on certain systems (reverts disabling resize) **Kevin One** - 03:42:52 PM	Files Modified 2 Lines Added 9 Lines Removed 2
MegaX8 * Fixed icon path issue. * Added includes.jar to .classpath * Notes: Build Path may need to be altered to OS dependent SWT. **Kevin One** - 05:04:07 AM	Files Modified 7 Lines Added 13 Lines Removed 9
MegaX7 * Added icon to title-bar/dock * Added report functionality (rudimentary) * Restructured report tab * Fixed time edit crash bug **Kevin One** - 01:22:38 AM	Files Modified 8 Lines Added 68 Lines Removed 14

2012-04-16

MegaX6 * Fixed missing column * Save to file support **Kevin One** - 04:29:12 AM	Files Modified 4 Lines Added 35 Lines Removed 41
Added a sample tsv **Kevin One** - 12:23:13 AM	Files Modified 1 Lines Added 6 Lines Removed 0
MegaX5 * Load from file works * Fixed delete crash bug * Fixed cell edit + select row bug * Fixed cell edit + delete bug * Fixed cell edit sanity check * Fixed table sort bug * Fixed a few other minor bugs * Code cleanup **Kevin One** - 12:09:52 AM	Files Modified 12 Lines Added 312 Lines Removed 393

2012-04-14

Minor fix to MegaX4.jar **Kevin One** - 10:49:01 PM	Files Modified 1 Lines Added 0 Lines Removed 0
MegaX4.jar * Merged Lucas' code * TaskObject.checkRecent * Column bidirectional sorting * Renamed columns * Edit notes (switch to tab1) * Edit time (inline cell edit with partial sanity check) * Clear button * cleanup **Kevin One** - 10:30:34 PM	Files Modified 8 Lines Added 293 Lines Removed 245
Worked in saved object adding comments and debugging **Lucas One** - 01:22:13 PM	Files Modified 5 Lines Added 107 Lines Removed 69

2012-04-11

End of class code **Lucas One** - 08:42:53 PM	Files Modified 5 Lines Added 151 Lines Removed 100
Added functionality for transferring time from stored tasks and added comments. **Lucas One** - 07:08:50 PM	Files Modified 52 Lines Added 4343 Lines Removed 4331
MegaX3.jar * Fixes for columns in Win OS * added recent list status in table. * UML diagram v2. **Kevin One** - 05:08:01 PM	Files Modified 8 Lines Added 20 Lines Removed 41
Buggy release MegaX2.jar **Kevin One** - 09:56:11 AM	Files Modified 17 Lines Added 329 Lines Removed 176
Created more functions to support task switching **Lucas One** - 08:10:39 AM	Files Modified 2 Lines Added 45 Lines Removed 10

2012-04-09

Changes to support saving to the task list **Lucas One** - 11:53:40 PM	Files Modified 14 Lines Added 258 Lines Removed 229

2012-04-05

classpath fix commit4 **Kevin One** - 07:25:39 AM	Files Modified 1 Lines Added 0 Lines Removed 1
classpath fix commit3 **Kevin One** - 07:16:59 AM	Files Modified 1 Lines Added 1 Lines Removed 0
classpath fix commit2 **Kevin One** - 07:14:27 AM	Files Modified 1 Lines Added 1 Lines Removed 0
classpath fix commit1 **Kevin One** - 07:10:58 AM	Files Modified 1 Lines Added 0 Lines Removed 1
MegaX1 push again to fix merged source files **Kevin One** - 07:08:04 AM	Files Modified 8 Lines Added 92 Lines Removed 109
Changes: * New: MegaX1.jar * Added untitled task behavior * Added file status label * Changed dialogs * Removed set to tab4 * Removed newTask spam protection * Removed force save file locks * Removed buttons * Cleanup **Kevin One** - 06:33:14 AM	Files Modified 9 Lines Added 24 Lines Removed 0

2012-04-04

UI/Feature strip down **Lucas One** - 09:41:35 PM	Files Modified 5 Lines Added 10 Lines Removed 130
final UI **Lucas One** - 08:04:17 PM	Files Modified 14 Lines Added 915 Lines Removed 924
fixed uml file **Kevin One** - 05:42:52 PM	Files Modified 1 Lines Added 0 Lines Removed 0
Added first UML diagram **Kevin One** - 05:31:17 PM	Files Modified 1 Lines Added 0 Lines Removed 0

2012-04-03

Several bug fixes. Pushing Megatype7 again. **Kevin One** - 02:11:16 PM	Files Modified 5 Lines Added 32 Lines Removed 19
* Megatype7 * Altered behavior * Added button states * Added listeners * Bug fixes * Cleanup **Kevin One** - 04:10:55 AM	Files Modified 23 Lines Added 1799 Lines Removed 1699

2012-04-02

* Megatype6.jar - Main.java, BrowsePath.java, Hooks.java, StopWatch.java, TableListener.java, TaskObject.java, Tools.java. **Kevin One** - 05:23:19 AM	Files Modified 29 Lines Added 2301 Lines Removed 454

2012-04-01

rebuilt jar with JRE v1.6 compat **Kevin One** - 08:34:35 PM	Files Modified 1 Lines Added 0 Lines Removed 0
Lucas' April 1st prototype **Kevin One** - 08:15:03 PM	Files Modified 1 Lines Added 0 Lines Removed 0
GUI bug Fixes: Added scroll bars to the notes text field (previously added recent tasks scroll bar) **Lucas One** - 07:53:24 PM	Files Modified 1 Lines Added 7 Lines Removed 14
- Created a class called TaskObject to hold variables for each task. - Fixed a bug in the timer where the count would continue even when the display had been paused. - optimized the UI organization. - created saveTasktoList() to place the current task into an element in the TaskObject array. (currently need debugging) - Started creating backend for the New Task button and the recent task list. **Lucas One** - 07:31:22 PM	Files Modified 3 Lines Added 254 Lines Removed 71
another screenshot **Kevin One** - 07:12:31 PM	Files Modified 1 Lines Added 0 Lines Removed 0
added screenshots **Kevin One** - 07:07:44 PM	Files Modified 2 Lines Added 0 Lines Removed 0
just a test commit **Lucas One** - 06:34:27 PM	Files Modified 1 Lines Added 1 Lines Removed 1
Browse dialog, more hooks, and etc. **Kevin One** - 05:24:15 AM	Files Modified 23 Lines Added 1084 Lines Removed 70
Another March 32nd update **Kevin One** - 02:13:59 AM	Files Modified 1 Lines Added 0 Lines Removed 0
Update for March 32nd **Kevin One** - 01:53:17 AM	Files Modified 1 Lines Added 0 Lines Removed 0

2012-03-31

Documentation Kevin One - 06:15:55 PM	Files Modified 1 Lines Added 7 Lines Removed 0
Fixed buildpath. Kevin One - 06:09:57 PM	Files Modified 8 Lines Added 152 Lines Removed 152
test3 Kevin One - 06:07:14 PM	Files Modified 1 Lines Added 3 Lines Removed 0
not quite right Kevin One - 06:05:22 PM	Files Modified 2 Lines Added 7 Lines Removed 1
test2 Kevin One - 06:02:18 PM	Files Modified 1 Lines Added 0 Lines Removed 7
Lets try this again Kevin One - 05:59:46 PM	Files Modified 7 Lines Added 15 Lines Removed 6
Test push Kevin One - 05:54:01 PM	Files Modified 24 Lines Added 4 Lines Removed 0
Add ignore rule Kevin One - 05:51:31 PM	Files Modified 1 Lines Added 1 Lines Removed 0
Removed platform-specific jars from build path Kevin One - 05:49:17 PM	Files Modified 1 Lines Added 0 Lines Removed 6

2012-03-30

project mockup APNGs **Kevin One** - 09:18:28 PM	Files Modified 2 Lines Added 0 Lines Removed 0
Bug fix - close gracefully. **Kevin One** - 08:19:21 PM	Files Modified 3 Lines Added 3 Lines Removed 0
Some clock fun. * + sets the clock to 25 seconds. * - reset to 0 seconds elapsed. Purely proof of concept work. **Kevin One** - 01:52:39 AM	Files Modified 8 Lines Added 82 Lines Removed 5

2012-03-29

Added cross-platform SWT jar **Kevin One** - 11:36:35 AM	Files Modified 1 Lines Added 0 Lines Removed 0
Added WindowBuilder support **Kevin One** - 11:29:03 AM	Files Modified 8 Lines Added 967 Lines Removed 92
SWING purge **Kevin One** - 10:00:26 AM	Files Modified 34 Lines Added 954 Lines Removed 680

2012-03-27

Minor cleanup **Kevin One** - 02:40:29 AM	Files Modified 1 Lines Added 0 Lines Removed 0
More SWT cleanup **Kevin One** - 02:23:24 AM	Files Modified 23 Lines Added 61 Lines Removed 516
Added cross-platform SWT support **Kevin One** - 02:01:23 AM	Files Modified 8 Lines Added 15 Lines Removed 0

2012-03-26

fix image **Kevin One** - 04:58:23 AM	Files Modified 1 Lines Added 0 Lines Removed 0
update image **Kevin One** - 04:48:54 AM	Files Modified 1 Lines Added 0 Lines Removed 0
rename wiki images **Kevin One** - 04:38:35 AM	Files Modified 4 Lines Added 0 Lines Removed 0
wiki images **Kevin One** - 04:35:48 AM	Files Modified 3 Lines Added 0 Lines Removed 0

2012-03-25

wiki image **Kevin One** - 09:05:02 PM	Files Modified 1 Lines Added 0 Lines Removed 0

2012-03-21

GUI work **Lucas One** - 07:38:43 PM	Files Modified 1 Lines Added 38 Lines Removed 0
Required build path jars **Lucas One** - 07:04:39 PM	Files Modified 13 Lines Added 12 Lines Removed 12
try 2 **Lucas One** - 06:55:52 PM	Files Modified 2 Lines Added 1 Lines Removed 0
supporting files for SWT interface **Lucas One** - 06:41:17 PM	Files Modified 4 Lines Added 620 Lines Removed 7
added code ref **Kevin One** - 02:06:13 PM	Files Modified 1 Lines Added 3 Lines Removed 0
Java filename standard compliance **Kevin One** - 12:31:43 PM	Files Modified 32 Lines Added 819 Lines Removed 760
jar #7 with stopwatch functionality **Kevin One** - 06:55:34 AM	Files Modified 1 Lines Added 0 Lines Removed 0
stopwatch attempt #4, success! **Kevin One** - 06:50:45 AM	Files Modified 15 Lines Added 200 Lines Removed 91

2012-03-19

stopwatch attempt #2 **Kevin One** - 10:48:08 PM	Files Modified 6 Lines Added 244 Lines Removed 4

2012-03-15

Added another JAR file to the cupboard. **Kevin One** - 03:19:50 AM	Files Modified 1 Lines Added 0 Lines Removed 0

2012-03-14

Added session persistence To-do: config file-based persistence **Kevin One** - 05:08:42 PM	Files Modified 3 Lines Added 56 Lines Removed 6

2012-03-13

Prototype5: * File I/O (create file and read) * Browse directory widget **Kevin One** - 05:45:43 AM	Files Modified 5 Lines Added 84 Lines Removed 6

2012-03-12

File IO test **Kevin One** - 05:14:26 AM	Files Modified 1 Lines Added 52 Lines Removed 0
File IO research **Kevin One** - 05:13:22 AM	Files Modified 1 Lines Added 12 Lines Removed 0

2012-03-11

added screenshot **Kevin One** - 11:07:01 PM	Files Modified 1 Lines Added 0 Lines Removed 0
Fixed JAR compatibility **Kevin One** - 11:03:23 PM	Files Modified 4 Lines Added 217 Lines Removed 217

2012-03-10

Added appendix info **Kevin One** - 10:01:31 PM	Files Modified 2 Lines Added 45 Lines Removed 0
Prototype3: * Added event handlers * Added JAR file **Kevin One** - 07:48:34 PM	Files Modified 10 Lines Added 173 Lines Removed 30
prototype screenshots **Kevin One** - 03:18:50 AM	Files Modified 1 Lines Added 0 Lines Removed 0
Prototype 2: * OOP: split project into many source files * GUI improvements * design progression * cleanup * JAR file **Kevin One** - 03:15:37 AM	Files Modified 15 Lines Added 328 Lines Removed 3

2012-03-09

Added prototype1 jar **Kevin One** - 04:29:21 AM	Files Modified 1 Lines Added 0 Lines Removed 0
Improved prototype **Kevin One** - 04:26:35 AM	Files Modified 1 Lines Added 78 Lines Removed 14
* Garbage cleanup * Played with several container types * Tab test with images * New prototype based on javax.swing.BoxLayout * New jar file * Some concrete progress **Kevin One** - 03:57:09 AM	Files Modified 16 Lines Added 407 Lines Removed 52

2012-03-07

another push attempt **Kevin One** - 06:54:58 PM	Files Modified 19 Lines Added 0 Lines Removed 0
Test Lucas **Lucas One** - 06:13:13 PM	Files Modified 2 Lines Added 1 Lines Removed 1
Mario too **Mario One** - 06:06:01 PM	Files Modified 2 Lines Added 1 Lines Removed 1
Mario test **Mario One** - 06:04:24 PM	Files Modified 2 Lines Added 1 Lines Removed 1

2012-03-04

Let's try this again **Kevin One** - 01:58:24 AM	Files Modified 1 Lines Added 5 Lines Removed 2
If you can see this, edit README to confirm **Kevin One** - 01:57:29 AM	Files Modified 2 Lines Added 2 Lines Removed 2
Added mockup 0 **Lucas One** - 01:53:05 AM	Files Modified 1 Lines Added 0 Lines Removed 0
created mockup dir **Kevin One** - 01:31:31 AM	Files Modified 0 Lines Added 0 Lines Removed 0
created mockup dir **Kevin One** - 01:30:05 AM	Files Modified 0 Lines Added 0 Lines Removed 0

2012-03-03

Minor cleanup **Kevin One** - 09:56:14 PM	Files Modified 2 Lines Added 1 Lines Removed 6
Anon tweaks **Kevin One** - 09:54:25 PM	Files Modified 1 Lines Added 1 Lines Removed 1
Increased frame width **Kevin One** - 09:48:19 PM	Files Modified 2 Lines Added 1 Lines Removed 1

A.8 Licensing

A.8.1 Dual-licensing

TimeMe version 1.0 is an open source project licensed under the Apache 2.0 license as well as the GNU GPL 3.0 license.

A.8.2 Apache 2.0 License

A.8.3 GPL 3.0 License

TimeMe – Java-based Task Tracker

Copyright (C) 2012 Team 0x00000001

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.