

Calendar Application: Prolog

COMP 4400: Principles of Programming Languages

Nandini Patel and Het Patel

Final Project Documentation

Calendar is a prolog based application that records all the daily tasks entered by the user. These records then can be used by the user itself to know at what time they have free time and the different tasks at different times. Through this program the user can add a task with a date, time, and, the name of the task into the record list.

The other functionality that has been added to this project is the example of the booking an appointment system. A user can check for an available appointment of any workplace. If there is no available record of the specified date and time by the user, the user can then query into prolog and make an appointment at that time.

To sum up the following is the list of all the functionalities:

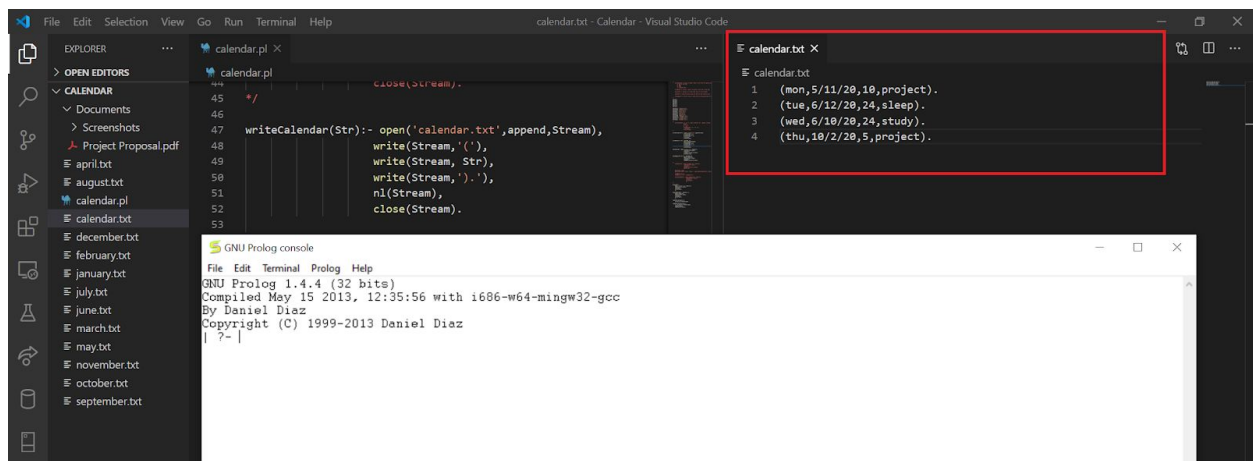
- 1) User can add a task to the calendar (calendar.txt file where all the tasks are stored)
- 2) Users can add tasks to their desired month (i.e. january.txt)
- 3) User can see all of their tasks in the general calendar (display)
- 4) User can see all of their tasks in the a specific month's calendar (display)
- 5) User can check all the appointment based on the date and time enter they enter
- 6) User can book an appointment after knowing that the time slot is available

Here we had a small complication where there is no matching result found it throws an exception on the console.

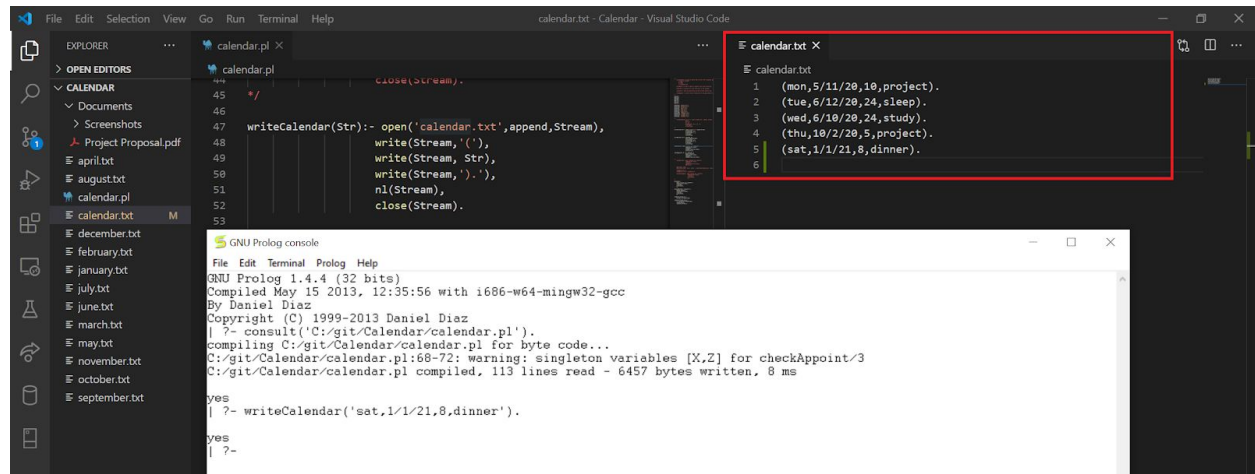
Our application has been explained in more depth using the screenshots below:

- 1) Add task to the task list

Before writing the query to add task:



After writing the query to add task:



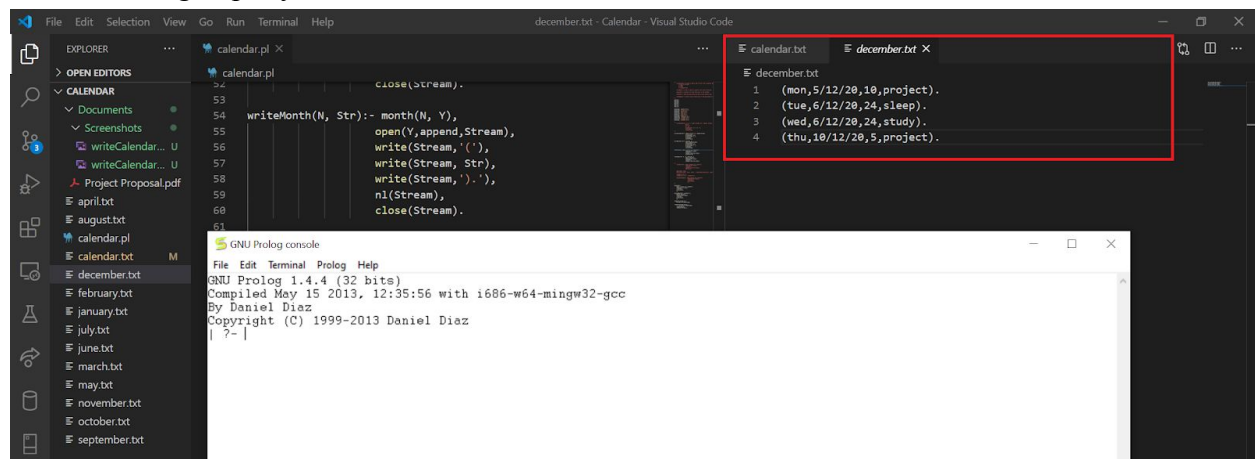
```
calendar.pl
44 close(Stream).
45
46
47 writeCalendar(Str):- open('calendar.txt',append,Stream),
48                       write(Stream,' '),
49                       write(Stream,Str),
50                       write(Stream,' '),
51                       nl(Stream),
52                       close(Stream).
53
```

```
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?- consult('C:/git/Calendar/calendar.pl').
compiling C:/git/Calendar/calendar.pl for byte code...
C:/git/Calendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppt/3
C:/git/Calendar/calendar.pl compiled, 113 lines read - 6457 bytes written, 8 ms

yes
| ?- writeCalendar('sat,1/1/21,8,dinner').

yes
| ?-
```

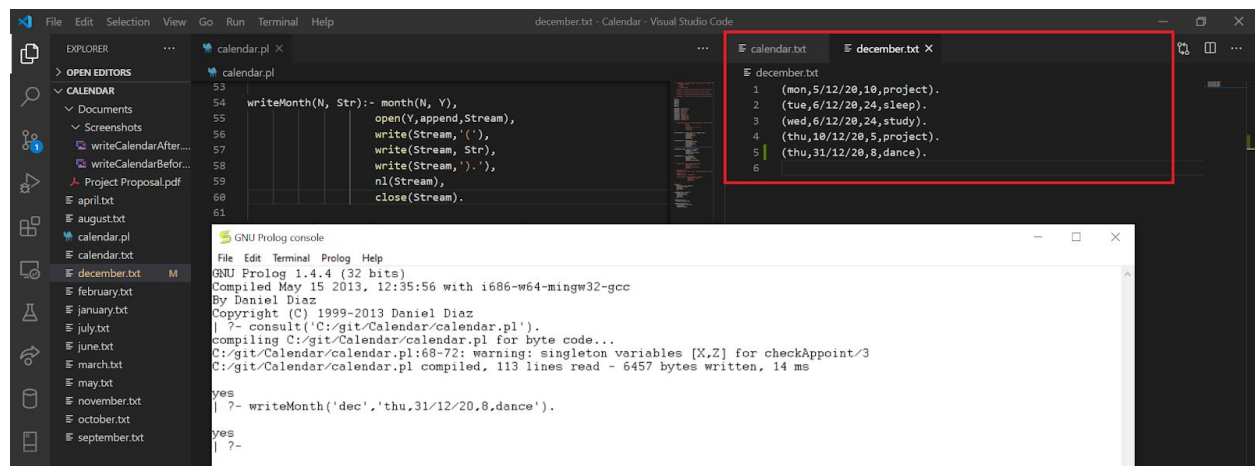
2) Add the task based on the month entered by the user to that specific month file
Before writing a query to a month:



```
calendar.pl
54 close(Stream).
55
56 writeMonth(N, Str):- month(N, Y),
57                      open(Y,append,Stream),
58                      write(Stream,' '),
59                      write(Stream,Str),
60                      write(Stream,' '),
61                      nl(Stream),
62                      close(Stream).
63
```

```
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?-
|
```

After writing the query to a month:



```
calendar.pl
54 close(Stream).
55
56 writeMonth(N, Str):- month(N, Y),
57                      open(Y,append,Stream),
58                      write(Stream,' '),
59                      write(Stream,Str),
60                      write(Stream,' '),
61                      nl(Stream),
62                      close(Stream).
63
```

```
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?- consult('C:/git/Calendar/calendar.pl').
compiling C:/git/Calendar/calendar.pl for byte code...
C:/git/Calendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppt/3
C:/git/Calendar/calendar.pl compiled, 113 lines read - 6457 bytes written, 14 ms

yes
| ?- writeMonth('dec', 'thu,31/12/20,8,dance').

yes
| ?-
```

3) Displays all the tasks that the user has assigned himself.

```
calendar.pl
93 displayAll:-
94   open('calendar.txt',read,Str),
95   readTasks(Str,Tasks),
96   close(Str),
97   write(Tasks),
98   nl.
99
100
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?- consult('C:/git/Calendar/calendar.pl').
compiling C:/git/Calendar/calendar.pl for byte code...
C:/git/Calendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppt/3
C:/git/Calendar/calendar.pl compiled, 113 lines read - 6457 bytes written, 8 ms
yes
| ?- displayAll.
[(mon,5/11/20,10,project),(tue,6/12/20,24,sleep),(wed,6/10/20,24,study),(thu,10/2/20,5,project),(sat,1/1/21,8,dinner)]
true ?
```

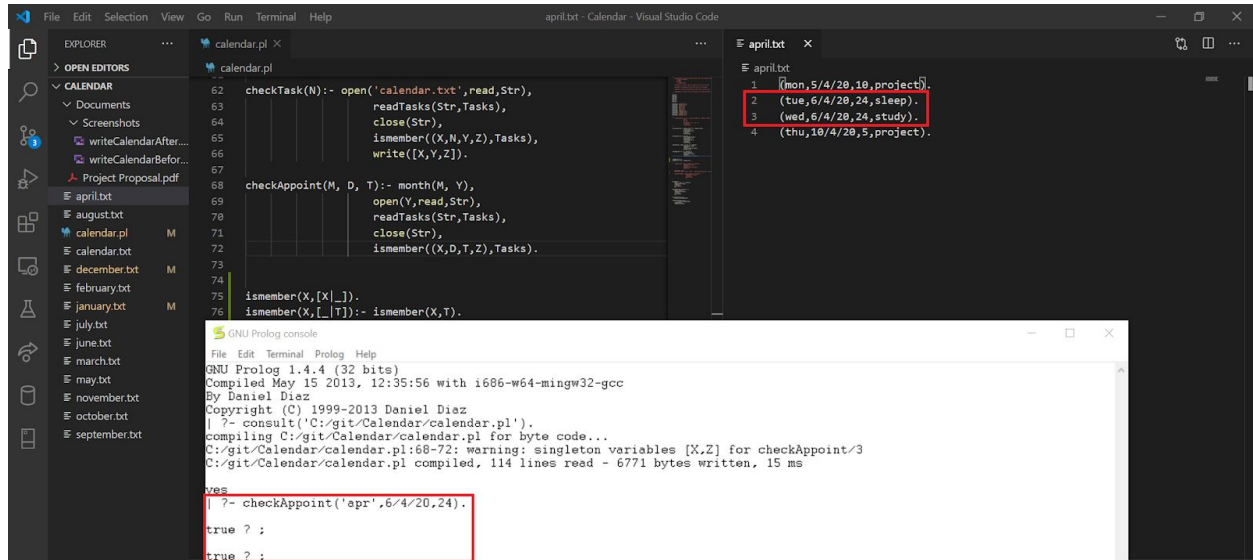
4) Using the function displayMonth, the user can display all the tasks in that specific month.

```
calendar.pl
100 displayMonth(M):- month(M, Y),
101   open(Y,read,Str),
102   readTasks(Str,Tasks),
103   close(Str),
104   write(Tasks),
105   nl.
106
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?- consult('C:/git/Calendar/calendar.pl').
compiling C:/git/Calendar/calendar.pl for byte code...
C:/git/Calendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppt/3
C:/git/Calendar/calendar.pl compiled, 113 lines read - 6457 bytes written, 10 ms
yes
| ?- displayMonth('jan').
[(mon,5/1/20,10,project),(tue,6/1/20,24,sleep),(wed,6/1/20,24,study),(thu,10/1/20,5,eat)]
true ?
```

5) Users can use the checkTask function to determine the tasks they have to do on that date.

```
calendar.pl
62 checkTask(N):- open('calendar.txt',read,Str),
63   readTasks(Str,Tasks),
64   close(Str),
65   ismember(X,N,Y,Z),Tasks,
66   write([X,Y,Z]).
67
68 checkAppt(M, D, T):- month(M, Y),
69   open(Y,read,Str),
70   readTasks(Str,Tasks),
71   close(Str),
72   ismember([X,D,T,Z],Tasks).
73
74
75 ismember(X,[X,_,_,_]).
76 ismember(X,[_,T,_,_]):- ismember(X,T).
77
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
| ?- consult('C:/git/Calendar/calendar.pl').
compiling C:/git/Calendar/calendar.pl for byte code...
C:/git/Calendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppt/3
C:/git/Calendar/calendar.pl compiled, 114 lines read - 6771 bytes written, 15 ms
yes
| ?- checkTask(6/12/20).
[tue,24,sleep]
true ?
```

6) checkAppoint is a function that is similar to checkTask. It differs in its output. The user can use this function to determine if the specific time entered them already exist in the list of appointments. If the output is true, then it means there exists a task or tasks at that given time.



The screenshot displays the Visual Studio Code interface with the 'calendar.pl' file open. The code defines two functions: `checkTask` and `checkAppoint`. The `checkAppoint` function is designed to check if a specific date and time already exist in the list of appointments. The file explorer on the left shows a directory structure for a calendar application, including files for each month and a main calendar file. The GNU Prolog console at the bottom shows the compilation of the code and the execution of a query: `?- checkAppoint('apr',6/4/20,24).`, which returns `true`, indicating that an appointment exists for that date and time.

```
calendar.pl
62 checkTask(N):- open('calendar.txt',read,Str),
63                readTasks(Str,Tasks),
64                close(Str),
65                ismember((X,N,Y,Z),Tasks),
66                write([X,Y,Z]).
67
68 checkAppoint(M, D, T):- month(M, Y),
69                        open(Y,read,Str),
70                        readTasks(Str,Tasks),
71                        close(Str),
72                        ismember((X,D,T,Z),Tasks).
73
74
75 ismember(X,[X|_]).
76 ismember(X,[_|_]):- ismember(X,T).
```

```
GNU Prolog console
File Edit Terminal Prolog Help
GNU Prolog 1.4.4 (32 bits)
Compiled May 15 2013, 12:35:56 with i686-w64-mingw32-gcc
By Daniel Diaz
Copyright (C) 1999-2013 Daniel Diaz
[ ?- consult('C:/git/Celendar/calendar.pl').
compiling C:/git/Celendar/calendar.pl for byte code...
C:/git/Celendar/calendar.pl:68-72: warning: singleton variables [X,Z] for checkAppoint/3
C:/git/Celendar/calendar.pl compiled, 114 lines read - 6771 bytes written, 15 ms

yes
[ ?- checkAppoint('apr',6/4/20,24).
true ? ;
true ? ;
```