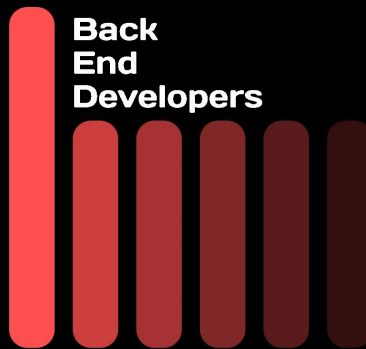


Module Interface Specification for Mechatronics Engineering



Team #1, Back End Developers

Jessica Bae

Oliver Foote

Jonathan Hai

Anish Rangarajan

Nish Shah

Labeeb Zaker

April 5, 2023

1 Revision History

Date	Version	Notes
2023-01-18	1.0	Initial documentation
2023-03-15	2.0	Minor improvements and proof reading for revision 1
2023-04-03	2.1	Incorporated TA feedback
2023-04-03	2.2	Included logo and added style to the document

2 Symbols, Abbreviations and Acronyms

Please refer to the System Requirements Specifications document at [this link](#) for relevant symbols, abbreviations.

Contents

1	Revision History	i
2	Symbols, Abbreviations and Acronyms	ii
3	Introduction	1
4	Notation	1
5	Module Decomposition	2
6	MIS of Battery Module	3
6.1	Module	3
6.2	Uses	3
6.3	Syntax	3
6.3.1	Exported Constants	3
6.3.2	Exported Access Programs	3
6.4	Semantics	3
6.4.1	State Variables	3
6.4.2	Environment Variables	3
6.4.3	Assumptions	3
6.4.4	Access Routine Semantics	4
6.4.5	Local Functions	4
7	MIS of microSD Module	5
7.1	Module	5
7.2	Uses	5
7.3	Syntax	5
7.3.1	Exported Constants	5
7.3.2	Exported Access Programs	5
7.4	Semantics	5
7.4.1	State Variables	5
7.4.2	Environment Variables	5
7.4.3	Assumptions	5
7.4.4	Access Routine Semantics	5
7.4.5	Local Functions	6
8	MIS of Local Database Module	7
8.1	Module	7
8.2	Uses	7
8.3	Syntax	7
8.3.1	Exported Constants	7
8.3.2	Exported Access Programs	7
8.4	Semantics	7

8.4.1	State Variables	7
8.4.2	Environment Variables	8
8.4.3	Assumptions	8
8.4.4	Access Routine Semantics	8
8.4.5	Local Functions	8
9	MIS of Reading Sensor Module	9
9.1	Module	9
9.2	Uses	9
9.3	Syntax	9
9.3.1	Exported Constants	9
9.3.2	Exported Access Programs	9
9.4	Semantics	9
9.4.1	State Variables	9
9.4.2	Environment Variables	9
9.4.3	Assumptions	10
9.4.4	Access Routine Semantics	10
9.4.5	Local Functions	10
10	MIS of Sensor Data Processing Module	11
10.1	Module	11
10.2	Uses	11
10.3	Syntax	11
10.3.1	Exported Constants	11
10.3.2	Exported Access Programs	11
10.4	Semantics	11
10.4.1	State Variables	11
10.4.2	Environment Variables	11
10.4.3	Assumptions	11
10.4.4	Access Routine Semantics	11
10.4.5	Local Functions	12
11	MIS of Prompt Validity Module	13
11.1	Module	13
11.2	Uses	13
11.3	Syntax	13
11.3.1	Exported Constants	13
11.3.2	Exported Access Programs	13
11.4	Semantics	13
11.4.1	State Variables	13
11.4.2	Environment Variables	13
11.4.3	Assumptions	13
11.4.4	Access Routine Semantics	13
11.4.5	Local Functions	13

12 MIS of Display System Module	14
12.1 Module	14
12.2 Uses	14
12.3 Syntax	14
12.3.1 Exported Constants	14
12.3.2 Exported Access Programs	14
12.4 Semantics	14
12.4.1 State Variables	14
12.4.2 Environment Variables	14
12.4.3 Assumptions	14
12.4.4 Access Routine Semantics	14
12.4.5 Local Functions	14
13 MIS of Prompt Generation Module	15
13.1 Module	15
13.2 Uses	15
13.3 Syntax	15
13.3.1 Exported Constants	15
13.3.2 Exported Access Programs	15
13.4 Semantics	15
13.4.1 State Variables	15
13.4.2 Environment Variables	15
13.4.3 Assumptions	15
13.4.4 Access Routine Semantics	15
13.4.5 Local Functions	15
14 MIS of Real Time Clock Module	16
14.1 Module	16
14.2 Uses	16
14.3 Syntax	16
14.3.1 Exported Constants	16
14.3.2 Exported Access Programs	16
14.4 Semantics	16
14.4.1 State Variables	16
14.4.2 Environment Variables	16
14.4.3 Assumptions	16
14.4.4 Access Routine Semantics	16
14.4.5 Local Functions	16
15 MIS of Create New User Module	17
15.1 Module	17
15.2 Uses	17
15.3 Syntax	17
15.3.1 Exported Constants	17
15.3.2 Exported Access Programs	17

15.4	Semantics	17
15.4.1	State Variables	17
15.4.2	Environment Variables	17
15.4.3	Assumptions	17
15.4.4	Access Routine Semantics	17
15.4.5	Local Functions	17
16	MIS of Configuration Module	18
16.1	Module	18
16.2	Uses	18
16.3	Syntax	18
16.3.1	Exported Constants	18
16.3.2	Exported Access Programs	18
16.4	Semantics	18
16.4.1	State Variables	18
16.4.2	Environment Variables	18
16.4.3	Assumptions	18
16.4.4	Access Routine Semantics	18
16.4.5	Local Functions	18
17	MIS of Graph Plotter	19
17.1	Module	19
17.2	Uses	19
17.3	Syntax	19
17.3.1	Exported Constants	19
17.3.2	Exported Access Programs	19
17.4	Semantics	19
17.4.1	State Variables	19
17.4.2	Environment Variables	19
17.4.3	Assumptions	19
17.4.4	Access Routine Semantics	19
17.4.5	Local Functions	19
18	MIS of Watch Strap Module	20
18.1	Module	20
18.2	Uses	20
18.3	Syntax	20
18.3.1	Exported Constants	20
18.3.2	Exported Access Programs	20
18.4	Semantics	20
18.4.1	State Variables	20
18.4.2	Environment Variables	20
18.4.3	Assumptions	20
18.4.4	Access Routine Semantics	20
18.4.5	Local Functions	20

19 MIS of Watch Case Module	21
19.1 Module	21
19.2 Uses	21
19.3 Syntax	21
19.3.1 Exported Constants	21
19.3.2 Exported Access Programs	21
19.4 Semantics	21
19.4.1 State Variables	21
19.4.2 Environment Variables	21
19.4.3 Assumptions	21
19.4.4 Access Routine Semantics	21
19.4.5 Local Functions	21

3 Introduction

The following document details the Module Interface Specifications for the EMAnator; the system currently being developed by the Back End Developers designed to aid in Ecological Momentary Assessment research. This document describes the various relevant details of interfacing with each module. These details include module descriptions, the uses of each module, the syntax of each module, and the semantics associated with each module.

Complementary documents include the System Requirement Specifications and the Module Guide. The Back End Developers highly recommend a thorough read-through of each document prior to a reading of this document to attain the prerequisite knowledge necessary to fully understand this MIS. The System Requirements Specifications can be found at [this link](#), and the Module Guide can be found at [this link](#).

4 Notation

The structure of the MIS for modules comes from [Hoffman and Strooper \(1995\)](#), with the addition that template modules have been adapted from [Ghezzi et al. \(2003\)](#). The mathematical notation comes from Chapter 3 of [Hoffman and Strooper \(1995\)](#). For instance, the symbol $:=$ is used for a multiple assignment statement and conditional rules follow the form $(c_1 \Rightarrow r_1 | c_2 \Rightarrow r_2 | \dots | c_n \Rightarrow r_n)$.

The following table summarizes the primitive data types used by Mechatronics Engineering.

Data Type	Notation	Description
Character	char	A single symbol or digit
Integer	\mathbb{Z}	A number without a fractional component in $(-\infty, \infty)$
Natural number	\mathbb{N}	A number without a fractional component in $[1, \infty)$
Real	\mathbb{R}	Any number in $(-\infty, \infty)$

The specification of Mechatronics Engineering uses some derived data types: sequences, strings, and tuples. Sequences are lists filled with elements of the same data type. Strings are sequences of characters. Tuples contain a list of values, potentially of different types. In addition, Mechatronics Engineering uses functions, which are defined by the data types of their inputs and outputs. Local functions are described by giving their type signature followed by their specification.

5 Module Decomposition

The following table is taken directly from the Module Guide document for this project.

Level 1	Level 2	Level 3
Hardware-Hiding Module	Battery Management	Battery
	Data Storage	microSD Database
	Sensor Array	Sensor Reading Sensor Data Processing Sensor Prompt Validity
	Physical Design	Watch Straps Watch Case
Behaviour-Hiding Module	Display System	Display Screen
	Prompt Generation	Prompt Generation
	Real Time Clock	RTC
Software Decision Module	Parameter Selection	Create New User Configuration
	Data Processing	Graph Data Display

Table 1: Module Hierarchy

6 MIS of Battery Module

6.1 Module

Bat_Man

6.2 Uses

None.

6.3 Syntax

6.3.1 Exported Constants

- Minimum battery threshold for voltage
- Maximum battery limit for voltage
- Low charge indicator for 10% limit

6.3.2 Exported Access Programs

Name	In	Out	Exceptions
bed_getBatteryLev	Bat_Raw	\mathbb{Z}	BED_ERR_BAT

6.4 Semantics

6.4.1 State Variables

None.

6.4.2 Environment Variables

- battery level
- low_Flag

6.4.3 Assumptions

System responds instantaneously to changes in flags (exported constants).

6.4.4 Access Routine Semantics

`bed_getBatteryLev`:

- Transition: Extracts the current battery level
- Output: Integer percentage value of the battery level.
- Exception: `BED_ERR_BAT = Z`

6.4.5 Local Functions

- `bed_lowIndicator`: boolean

7 MIS of microSD Module

7.1 Module

microSD_Stor

7.2 Uses

Sensor Prompt Validity Module (Section 11), Real Time Clock Module (Section 14)

7.3 Syntax

7.3.1 Exported Constants

None.

7.3.2 Exported Access Programs

Name	In	Out	Exceptions
listDir	FS, const char*, uint8_t	dirname: string	BED_ERR_SD: \mathbb{Z}
createDir	FS, const char *	-	BED_ERR_SD: \mathbb{Z}
removeDir	FS, const char *	-	BED_ERR_SD: \mathbb{Z}

7.4 Semantics

7.4.1 State Variables

None.

7.4.2 Environment Variables

- fs: A text file.

7.4.3 Assumptions

- MicroSD card is formatted correctly.
- MicroSD card is inserted correctly.

7.4.4 Access Routine Semantics

None.

7.4.5 Local Functions

- readFile
- writeFile
- appendFile
- renameFile
- deleteFile

8 MIS of Local Database Module

8.1 Module

Database_Stor

8.2 Uses

microSD Module (Section 7)

8.3 Syntax

8.3.1 Exported Constants

- Size of the field: First Name = 25
- Size of the field: Last Name = 25
- Size of the field: Gender = 25
- Size of the field: Phone Number = 20
- Size of the field: Email = 25
- Size of the field: Address = 255
- Size of the field: Device model number = 50

8.3.2 Exported Access Programs

Name	In	Out	Exceptions
object.connect	-	Read_Data: string, integer	BED_ERR_DB: \mathbb{Z}
object.cursor	Write_Data: string, integer	BED_ERR_DB: \mathbb{Z}	
object.execute	Write_Data: string, integer	BED_ERR_DB: \mathbb{Z}	
object.commit	Write_Data: string, integer	BED_ERR_DB: \mathbb{Z}	
object.close	Write_Data: string, integer	BED_ERR_DB: \mathbb{Z}	

8.4 Semantics

8.4.1 State Variables

None.

8.4.2 Environment Variables

- First Name
- Last Name
- Gender
- Phone Number
- Email
- Address
- Device Model Number

8.4.3 Assumptions

None.

8.4.4 Access Routine Semantics

- `object.connect()`: Connects to the database.
- `object.cursor()`: Establishes connection between the database and the query.
- `object.execute()`:
- `object.commit()`:
- `object.close()`: Disconnects from the database.

8.4.5 Local Functions

None.

9 MIS of Reading Sensor Module

9.1 Module

Sensor_Reding

9.2 Uses

Battery Management (Section [6](#))

9.3 Syntax

9.3.1 Exported Constants

- ACCEL_SENSITIVITY: Z
- GYRO_SENSITIVITY: Z
- Threshold: Z
- MPU_CALIBRATION: Z

9.3.2 Exported Access Programs

- bed_mpu_detect
- bed_hr_detect

9.4 Semantics

9.4.1 State Variables

currTime : \mathbb{R}

9.4.2 Environment Variables

- curr_ax
- curr_ay
- curr_az
- curr_gx
- curr_gy
- curr_gz

9.4.3 Assumptions

- All activity thresholds are provided from the configuration file.

9.4.4 Access Routine Semantics

- `bed_mpu_detect`: Returns the current values of the accelerometer and gyroscope.
- `bed_hr_detect`: Returns the current values of heartrate sensor.

9.4.5 Local Functions

- `bed_mpu_setup`
- `bed_hr_setup`

10 MIS of Sensor Data Processing Module

10.1 Module

Sensor_Data

10.2 Uses

Sensor Reading (Section 9)

10.3 Syntax

10.3.1 Exported Constants

- ACTIVITY_STEPS
- ACTIVITY_IDLE_RESET
- ACTIVITY_IDLE_WAIT

10.3.2 Exported Access Programs

Name	In	Out	Exceptions
HeartRate_Read	-	Read_Data: integer	BED_ERR_SE: \mathbb{Z}
MPU_Read	-	Read_Data: integer	BED_ERR_SE: \mathbb{Z}
Touch_Read	-	Read_Data: bool	BED_ERR_SE: \mathbb{Z}

10.4 Semantics

10.4.1 State Variables

None.

10.4.2 Environment Variables

None.

10.4.3 Assumptions

- There is space available in microSD card.

10.4.4 Access Routine Semantics

None.

10.4.5 Local Functions

`data_smoothing_filter()`: The purpose of this function is to make sure that the data coming from all the sensors is smoothed, in order to prevent a prompt from being generated erroneously and disturbing the user.

11 MIS of Prompt Validity Module

11.1 Module

Sensor_Prompt

11.2 Uses

Sensor Data Processing (Section 10)

11.3 Syntax

11.3.1 Exported Constants

11.3.2 Exported Access Programs

Name	In	Out	Exceptions
Check_Validity	Read_Data	Prompt_Valid: bool	BED_ERR_SE: \mathbb{Z}

11.4 Semantics

11.4.1 State Variables

None.

11.4.2 Environment Variables

None.

11.4.3 Assumptions

None.

11.4.4 Access Routine Semantics

None.

11.4.5 Local Functions

12 MIS of Display System Module

12.1 Module

Disp_Sys

12.2 Uses

Prompt Generation Module (Section 13), Real Time Clock Module (Section 14), Battery Management (Section 6)

12.3 Syntax

12.3.1 Exported Constants

None.

12.3.2 Exported Access Programs

Name	In	Out	Exceptions
Disp_Time	-	-	BED_ERR_DISP: \mathbb{Z}
Disp_Prompt	Prompt: string	Response: string	BED_ERR_DISP: \mathbb{Z}
Switch_Window	Window: \mathbb{Z}	-	BED_ERR_DISP: \mathbb{Z}

12.4 Semantics

12.4.1 State Variables

None.

12.4.2 Environment Variables

None.

12.4.3 Assumptions

None.

12.4.4 Access Routine Semantics

None.

12.4.5 Local Functions

None.

13 MIS of Prompt Generation Module

13.1 Module

Prompt_Gen

13.2 Uses

Sensor Array Module (Section ??)

13.3 Syntax

13.3.1 Exported Constants

max_prompts : \mathbb{Z}

13.3.2 Exported Access Programs

Name	In	Out	Exceptions
Access_Prompt	Prompt_: \mathbb{Z}	Prompt: Struct	BED_ERR_PG: \mathbb{Z}

13.4 Semantics

13.4.1 State Variables

None.

13.4.2 Environment Variables

None.

13.4.3 Assumptions

None.

13.4.4 Access Routine Semantics

None.

13.4.5 Local Functions

None.

14 MIS of Real Time Clock Module

14.1 Module

RTC

14.2 Uses

None.

14.3 Syntax

14.3.1 Exported Constants

None.

14.3.2 Exported Access Programs

Name	In	Out	Exceptions
Get_DateTime	-	\mathbb{R}	BED_ERR_RTC: \mathbb{Z}

14.4 Semantics

14.4.1 State Variables

Date: string

Time: string

14.4.2 Environment Variables

None.

14.4.3 Assumptions

- Initial date and time is correctly set.

14.4.4 Access Routine Semantics

None.

14.4.5 Local Functions

None.

15 MIS of Create New User Module

15.1 Module

NewUser_Enter

15.2 Uses

Local Database Module (Section [12](#))

15.3 Syntax

15.3.1 Exported Constants

None.

15.3.2 Exported Access Programs

None.

15.4 Semantics

15.4.1 State Variables

param_input: string, integer

15.4.2 Environment Variables

File: A new information line in database.

15.4.3 Assumptions

- All configuration parameters within acceptable limits.

15.4.4 Access Routine Semantics

None.

15.4.5 Local Functions

None.

16 MIS of Configuration Module

16.1 Module

Config_Param

16.2 Uses

MicroSD Module (Section [7](#))

16.3 Syntax

16.3.1 Exported Constants

None.

16.3.2 Exported Access Programs

None.

16.4 Semantics

16.4.1 State Variables

param_input: string

16.4.2 Environment Variables

None.

16.4.3 Assumptions

- All configuration parameters within acceptable limits.

16.4.4 Access Routine Semantics

None.

16.4.5 Local Functions

None.

17 MIS of Graph Plotter

17.1 Module

Graph_Plot

17.2 Uses

Device Manager Module (Section ??)

17.3 Syntax

17.3.1 Exported Constants

None.

17.3.2 Exported Access Programs

None.

17.4 Semantics

17.4.1 State Variables

graph_select: bool

17.4.2 Environment Variables

File: A database on the host computer.

17.4.3 Assumptions

- Data is in proper format and not corrupted.

17.4.4 Access Routine Semantics

None.

17.4.5 Local Functions

graph_stat: Statistical analysis function.

18 MIS of Watch Strap Module

18.1 Module

Watch_Strap

18.2 Uses

Watch Case Module (Section [19](#))

18.3 Syntax

Velcro straps going tied to the watch case.

18.3.1 Exported Constants

None.

18.3.2 Exported Access Programs

None.

18.4 Semantics

Allow the device to be strapped onto the user.

18.4.1 State Variables

None.

18.4.2 Environment Variables

None.

18.4.3 Assumptions

Will not impact the functionality of any other components.

18.4.4 Access Routine Semantics

None.

18.4.5 Local Functions

None.

19 MIS of Watch Case Module

19.1 Module

Watch_Case

19.2 Uses

None.

19.3 Syntax

CAD model and 3D printed.

19.3.1 Exported Constants

None.

19.3.2 Exported Access Programs

None.

19.4 Semantics

Holds the display screen, touch bezels, and display screen together into 1 single device.

19.4.1 State Variables

None.

19.4.2 Environment Variables

None.

19.4.3 Assumptions

Will not impact the functionality of any other components.

19.4.4 Access Routine Semantics

None.

19.4.5 Local Functions

None.

References

Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli. *Fundamentals of Software Engineering*. Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2003.

Daniel M. Hoffman and Paul A. Strooper. *Software Design, Automated Testing, and Maintenance: A Practical Approach*. International Thomson Computer Press, New York, NY, USA, 1995. URL <http://citeseer.ist.psu.edu/428727.html>.