System Design Document

For

NASA VESTIBULAR CHAIR

Team member:
Brandon Boyle-Fagan
Matthaeus Gebauer
Miles Osborne
Dylan Prothro
Noah Reid
Kent Wilson

Version/Author	Date
1	10/6/2022
1.1	10/27/2022
1.2	11/8/22

TABLE OF CONTENT

SYSTEM DESIGN DOCUMENT

1 INTRODUCTION

1.1 Purpose and Scope

The goal of the NASA Vestibular Chair project is to restore the basic hardware functionality of a rotating chair that was acquired by Embry-Riddle that was used by NASA to test the human vestibular system and provide it with a new controller. This consists of ensuring the system can reach a specified RPM, hold that specified RPM for a set duration and allow the servos to gradually slow down to idle. Once the initial controller is created and able to provide intended the functionality to the vestibular char, we will begin integrating more modern software and hardware tools to improve the "quality of life" features of the chair. This includes a web interface, custom test profiles/sequences, and the ability to read and store sensor data from the chair.

1.2 Project Executive Summary

This section provides an overview of the NASA Vestibular Chair project from a macro perspective, showing the framework with which the system design was conceived.

1.2.1 System Overview

The NASA Vestibular Chair system consists of the hardware component of the chair itself, which is connected to a controller composed of a motor controller and other components to be able to handle digital and analog inputs for the chair motor. An additional web interface is planned to be added to the controller to allow for a more precise measurement of the input for the chair. The hardware components of the chair consist of a tachometer, motor, and the actual chair itself as well as some other pins set up for other once-used analog measurements. Figure 1 listed below, details a use case diagram of the NASA Vestibular chair system.

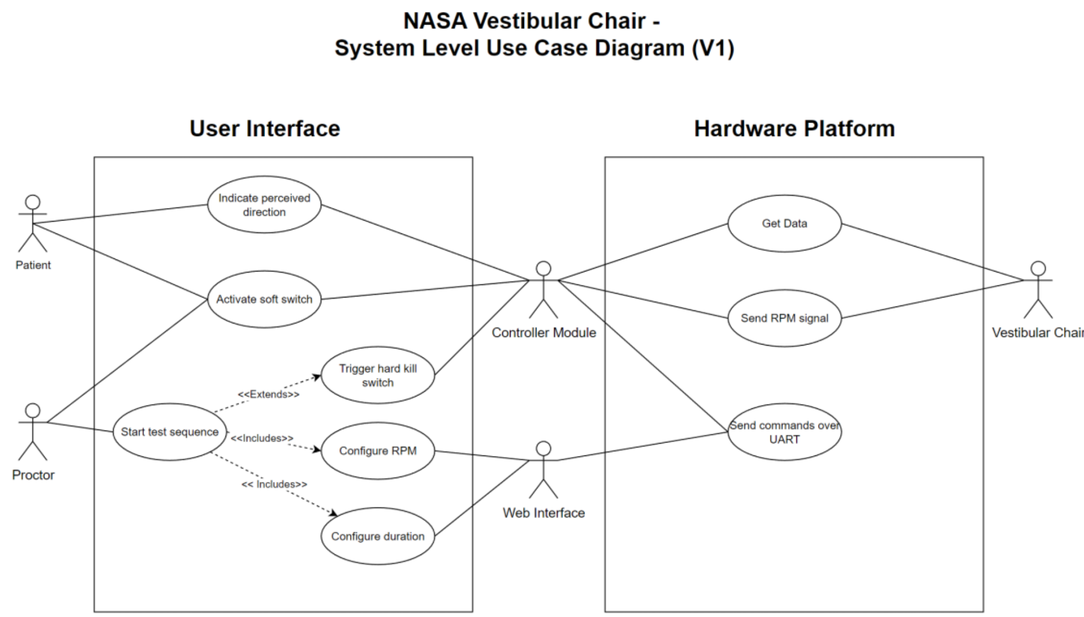

Figure 1: Use Case Diagram for NASA Vestibular Chair

1.2.2 Design Constraints

The development team plans to keep as much of the original internal hardware of the chair as possible. With this in mind, one of the major constraints in design of the controller of the chair and its interface is to have the new components work with the older technology present in the chair. Another constraint comes from the speed the chair is going to be allowed to move. The chair needs to be set to not move faster than a rotation of 100 degrees per second, and with that constraint, the controller needs to be programmed to not allow an input of voltage that would cause a rotational speed higher than that. The controller also needs to have a mechanical kill switch, meaning it needs to be connected to the power being sent to the chair via the controller to be able to shut off power being sent to the chair if necessary.

1.2.3 Future Contingencies

Future contingencies include the inclusion of a digital part of the controller to account for any analog issues, with the intent to keep an analog and digital component to control the inputs of the chair. Another contingency comes from the alternative components that the team researched in case the planned components are not available or do not work for the direction of the project. This includes researching two different types of motor controllers, Pololu - High-Power Simple Motor Controller G2 24v12 and Pololu - RoboClaw 2x15A Motor Controller (V5E). While the G2 is the preferred motor controller for the system, the functionality of the RoboClaw was investigated and was determined to be able to be used in place of the G2 if issues arise such as lack of power output or component damage. The inclusion of a mechanical kill switch is a contingency to avoid having the chair spin out of control if too high of a voltage is supplied to the device.

1.3 Document Organization

This System Design Document is organized into six sections. The introduction section explains the basics of the design, followed by the document going over the system architecture, the human-machine interface, the detailed design, and external interfaces.

1.4 Project References

No references at this time.

2 SYSTEM ARCHITECTURE

The system uses an interface of hardware and software to control the movement of the NASA Vestibular Chair, as well as measure its speed and other readings.

2.1 System Hardware Architecture

The composition of the hardware is represented by

- NASA chair
  - The NASA chair consists of inner hardware such as its motor and sensors already installed in the device such as a tachometer, which will be used to measure the speed of the chair during operation. The plan is to keep the sensors that are already in the chair if they are still usable, however it is also on the table to replace them if necessary.

- Controller Module
  - The controller module will consist of a dedicated microcontroller, internal motor controller, and physical I/O such as a potentiometer and switches to quickly control the state and execution of the system. The current plan is to use the Nucleo-WB55RG microcontroller to control the system and interface with the G2 24v12 motor controller to achieve the desired functionality. The motor controller has, upon initial inspection, seemed like one of the best choices for the overall scope of the project.

- Wireless bi-directional input
  - This module is a recent addition to the system. Part of the core functionality of the NASA vestibular chair is to allow the user to indicate what direction they believe to be spinning in. Due to the physical constraints of the chair spinning, we must use a wireless input to give the user this feature. However, instead of buying an off-the-shelf component, there is the argument for fabricating one. At the current state of development, this is not a high priority and this topic will be expanded in a later revision.

2.2 System Software Architecture

The software is planned to be an embedded systems project mostly developed using C++. Since the microcontroller selected is the Nucleo-WB55RG, the STM32 HAL library is available to use and officially supported. Other frameworks can be utilized such as the libopencm3 framework. Due to the urgency of the project, writing drivers for each component of the project is not feasible and would artificially and unnecessarily increase the difficulty of the project.

The Nucleo-WB55RG is the core of the controller module and will interface with several peripherals including the motor controller and two seven-segment displays. Both of these peripherals can be controlled via the I2C protocol which makes the software written easier to implement and maintain.

The second aspect of the system is the web interface which gives the proctor more options as it relates to creating test sequences and displaying data during the test. Data is read from the chair via the analog pins and then temporarily stored in a buffer. The data is then sent over UART to the web interface and displayed on the host computer. Likewise, when the proctor wants to actuate the chair, they will use the web interface to configure the test. When ready, the test sequence information will be sent via commands over UART and then processed by the Nucleo microcontroller. The system shall then begin to follow the sequence of events as indicated by the proctor’s designated test sequence.

A state transition chart for the System based excluding the web interface is listed below. The web interface was excluded at this time to focus on the current priority of more robust mechanical control. However, when the web interface and accompanying features are introduced, the state transition chart should be very similar.

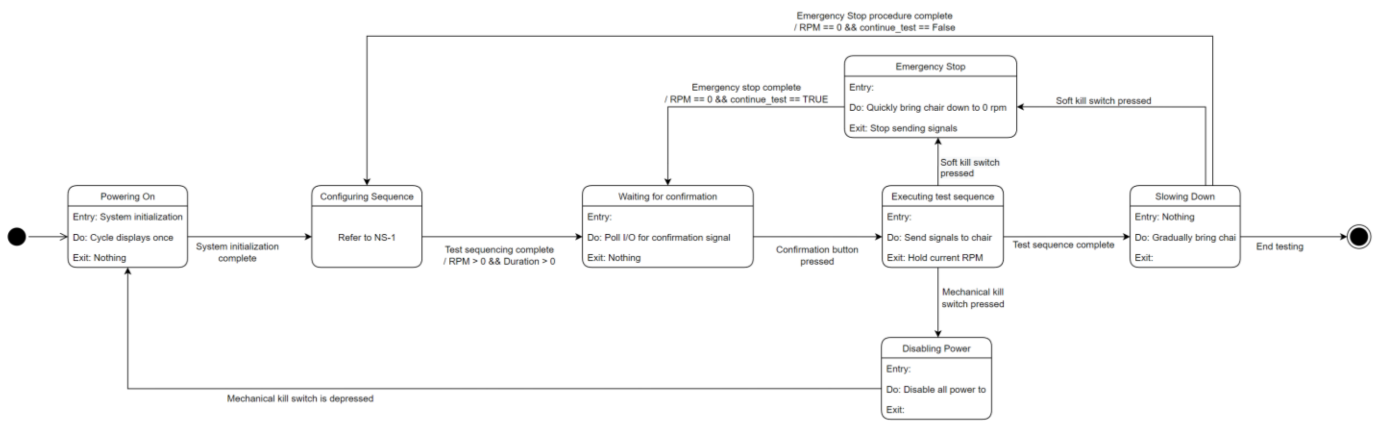

Figure 2: State Transition Chart for NASA Vestibular Chair

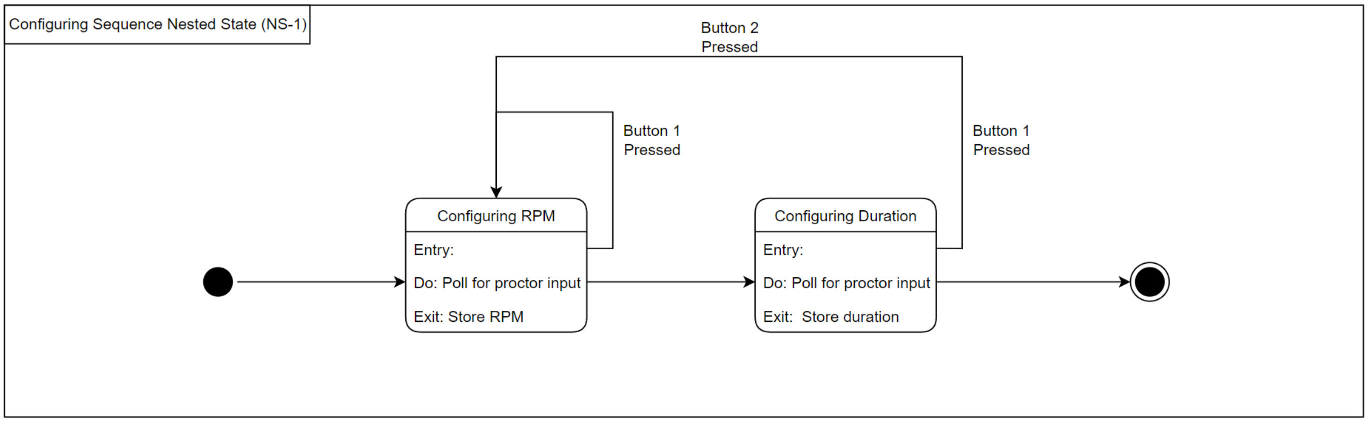

Figure 2.1: Nested State 1 for State Transition Chart (NS-1)

2.3 Internal Communications Architecture

In the NASA Vestibular Chair System, the main communication channels happen over UART from sending and receiving commands from the web interface and processing digital signals from the controller module I/O. Additionally, I2C will be utilized by the Nucleo microcontroller to interface with the motor controller and the seven-segment displays. Lastly, Bluetooth will be utilized to receive data from the wireless input device indicating what direction the user believes they are spinning in. This is feasible because the Nucleo-WB55RG has a built-in Bluetooth module. Due to the recent changes in the communication architecture, a diagram detailing the inter-communication between components is not ready at this time but will be included in the next revision of this document.

Figure 3 listed below is a simple data flow diagram, detailing the critical piece of information being transferred from components.To make the diagram more simple, the controller module is assumed to include all required electronics including the Nucleo Microcontroller, G2 motor controller, seven segment leds, switches, and the potentiometer.

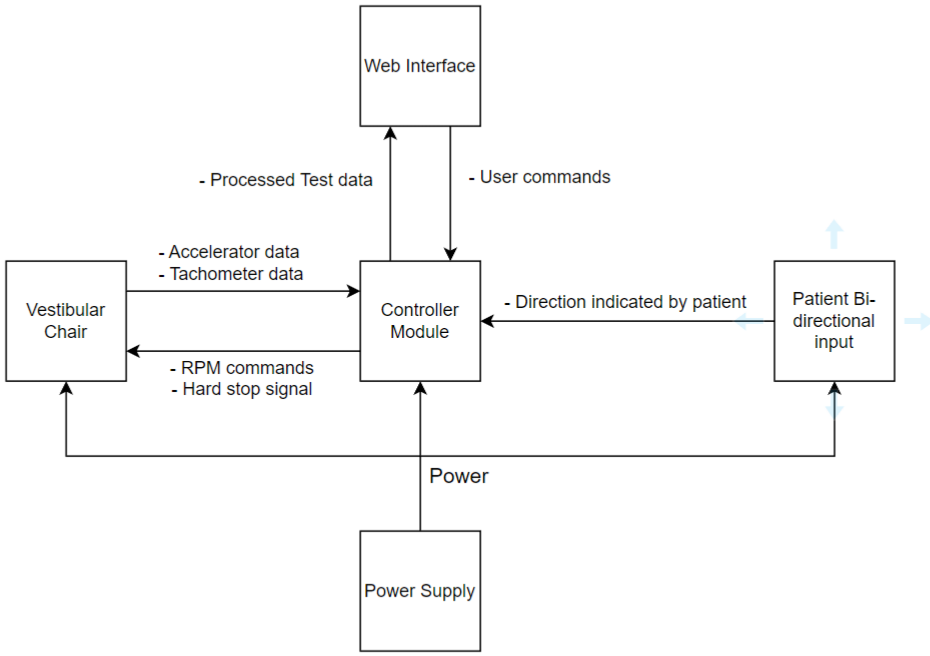

Figure 3: Level 0 Data Flow Diagram for NASA Vestibular Chair System

3 HUMAN-MACHINE INTERFACE

Human input will be necessary for the system as the input for the chair is needed to increase its voltage and by extension its speed. This interaction will cause the user to be able to adjust the chair's speed gradually through the controller device.

3.1 Inputs

The inputs of the system will be given from the controller to the chair to allow movement of the chair or set up of testing cases for the voltage to be applied without needing to be adjusted by the user. These inputs are user-generated, as the individual in control of the device sets the voltage that the chair is being given. The sensors used will also count as input, including things such as the tachometer and accelerometer. Another input will come from the individual sitting in the chair, as they will be able to send data indicating the direction, they believe they are spinning in.

3.2 Outputs

The output response from the chair should be the tachometer reading to feed into the controller to control the speed via its feedback. This will allow the system to avoid reaching a speed that outpaces the scope of the controller. The other output is that given from the chair itself, which is its actual rotation given the input voltage supplied by the controller. The feedback from the chair should also be able to communicate the direction it is rotating to the controller.

4 DETAILED DESIGN

This section contains detailed information about the hardware and software design of the system

4.1 Hardware Detailed Design

The system is centered around the Nucleo-WB55RG microcontroller. This component will perform the following roles at a minimum:

1. Accept user input from the proctor
2. Perform data acquisition at a sufficient sampling rate
3. Process commands sent via UART
4. Send data to peripherals through I2C
5. Process commands via Bluetooth from wireless bi-directional input
6. Send data to the web interface via UART
7. Control the vestibular chair's RPM via commands sent to the motor controller.

Figures 4, 5, and 6 are a simplified overview of the hardware and components present in the system. Note, that these diagrams do not document the internal communication between each module.

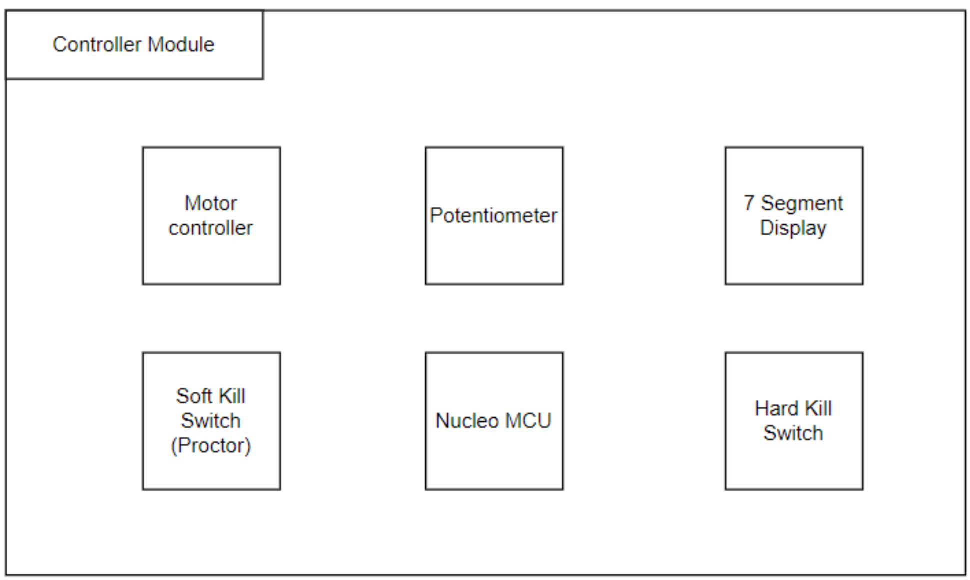

Figure 4: Simplified overview of NASA Vestibular Chair Controller Module

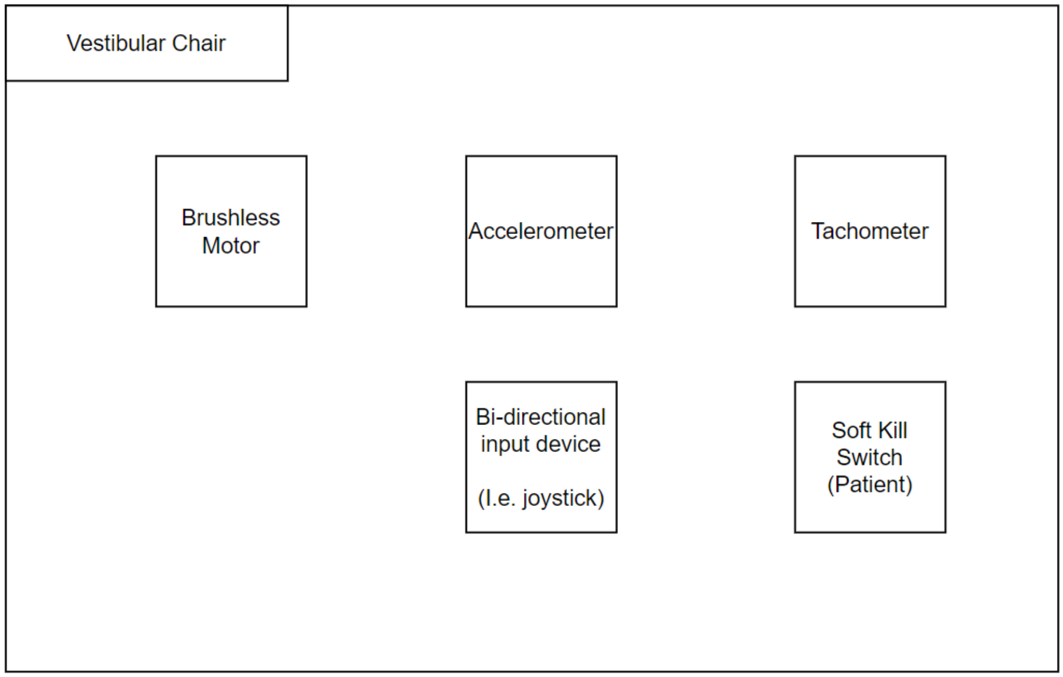

Figure 5: Overview of NASA Vestibular Chair Components

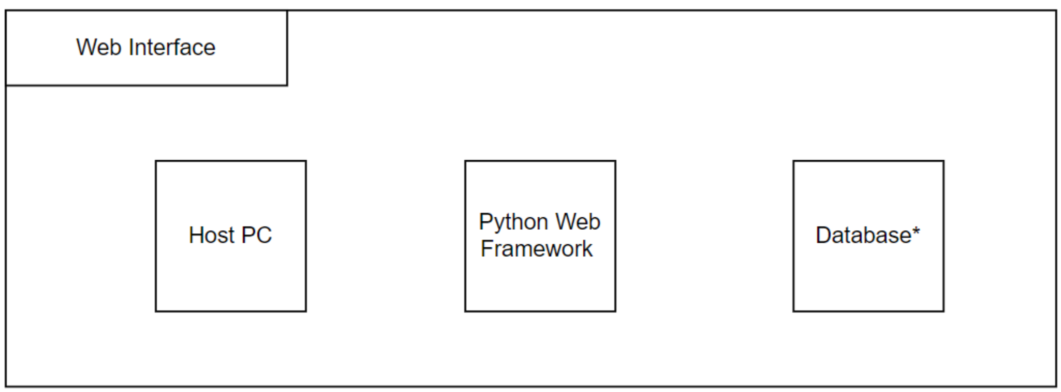

Figure 6: Overview of Web Interface Components

4.2 Software Detailed Design

A software module is the lowest level of design granularity in the system. Depending on the software development approach, there may be one or more modules per system. This section should provide enough detailed information about logic and data necessary to completely write source code for all modules in the system (and/or integrate COTS software programs).

If there are many modules or if the module documentation is extensive, place it in an appendix or reference a separate document. Add additional diagrams and information, if necessary, to describe each module, its functionality, and its hierarchy. Industry-standard module specification practices should be followed. Include the following information in the detailed module designs:

- A narrative description of each module, its function(s), the conditions under which it is used (called or scheduled for execution), its overall processing, logic, interfaces to other modules, interfaces to external systems, security requirements, etc.; explain any algorithms used by the module in detail
- For COTS packages, specify any call routines or bridging programs to integrate the package with the system and/or other COTS packages (for example, Dynamic Link Libraries)
- Data elements, record structures, and file structures associated with module input and output
- Graphical representation of the module processing, logic, flow of control, and algorithms, using an accepted diagramming approach (for example, structure charts, action diagrams, flowcharts, etc.)
- Data entry and data output graphics; define or reference associated data elements; if the project is large and complex or if the detailed module designs will be incorporated into a separate document, then it may be appropriate to repeat the screen information in this section
- Report layout

5 EXTERNAL INTERFACES

The current external interface is the idea/plan to add a web based component for the controller. This is not currently within the scope of the basics of the system; however, it is something that is planned to be attempted time permitting.

5.1 Interface Architecture

In this section, describe the interface(s) between the system being developed and other systems; for example, batch transfers, queries, etc. Include the interface architecture(s) being implemented, such as wide area networks, gateways, etc. Provide a diagram depicting the communications path(s) between this system and each of the other systems, which should map to the context diagrams in Section 1.2.1. If appropriate, use subsections to address each interface being implemented.

5.2 Interface Detailed Design

For each system that provides information exchange with the system under development, there is a requirement for rules governing the interface. This section should provide enough detailed information about the interface requirements to correctly format, transmit, and/or receive data across the interface. Include the following information in the detailed design for each interface (as appropriate):

- The data format requirements; if there is a need to reformat data before they are transmitted or after incoming data is received, tools and/or methods for the reformat process should be defined
- Specifications for hand-shaking protocols between the two systems; include the content and format of the information to be included in the hand-shake messages, the

timing for exchanging these messages, and the steps to be taken when errors are identified
- Format(s) for error reports exchanged between the systems; should address the disposition of error reports; for example, retained in a file, sent to a printer, flag/alarm sent to the operator, etc.
- Graphical representation of the connectivity between systems, showing the direction of data flow
- Query and response descriptions

If a formal Interface Control Document (ICD) exists for a given interface, the information can be copied, or the ICD can be referenced in this section.

6 SYSTEM INTEGRITY CONTROLS

- Due to the nature of the interaction between the user and the chair itself, there is little question of integrity for the system due to it not being able to be used in a way that would cause a leak of sensitive information.
- The most likely issue would be misuse of the chair itself, such as spinning it at velocities it is not built for, however there will be a built in control and failsafe to avoid such outcomes.