Thermaquil user story

# Definitions

| Acronym/term | Definition |
| --- | --- |
| Therapy/thermal therapy | The treatment delivered to patients by the Thermaquil system. |
| Treatment | Unless otherwise specified, a synonym for Therapy/thermal therapy. |
| The pad | The patient-contacting part of the Thermaquil system |
| The system (a.k.a., the Thermaquil system or the device) | The Thermaquil hardware and embedded code used to heat, cool, and pump fluid used to deliver thermal therapy to the patient via the pad. |
| The App | The tablet/mobile application used to control the system. |
| Controller | The tablet/mobile device used to run the app |
| Cycle | A programmed time/temperature combination used for thermal therapy |
| Hot Cycle(s) | Cycles that use hot temperatures |
| Cold Cycle(s) | Cycles that use cold temperatures |
| Full Cycle(s) | A “Full Cycle” is one Hot Cycle followed by one Cold Cycle |
| Patient | The individual being treated with the Thermaquil system. |
| Patient’s profile | All patient-specific information stored by the System. Most notably, the default treatment program used for the second and subsequent treatments unless a replacement program is sent by the App. |
| User | The individual controlling the Thermaquil system to treat the Patient. The User is usually the Patient but may also be a caregiver, physician, or other clinical staff. |

# Core use cases

## First use

The first use of the device is typically in a physician’s office. The first treatment typically consists of 4+ heat/cold cycles and increases from moderate times/temperatures to increased intensity.

| Step | Device actions | App actions |
| --- | --- | --- |
| Patient checks in at the physician’s office |  |  |
| Patient fills out baseline forms with research coordinator |  | 1. App allows the patient/coordinator to fill out forms, sends results to Thermaquil’s AWS database |
| The user turns the device on for the first time | 1. Waits for app to authorize the user and to create a profile 2. Device connects to app with a secure Bluetooth connection using Serial Port Protocol. | 1. App arrives installed on a tablet that is paired with the device 2. Displays screen prompting passcode before allowing use for the first time 3. Patient enters passcode 4. App creates a secure Bluetooth connection 5. Device notifies patient that pairing is successful |
| Patient is prepared to start treatment | 1. Waits for commands from the App 2. Receives command from App to prepare for treatment 3. Prepares for treatment 4. Notifies the App when reservoir temperatures are ready to begin treatment. | 1. Sends first Full Cycle information to the System and sends a command to prepare for treatment. Example first cycle information:  | Hot temp | Hot time | Cold temp | Cold time | | --- | --- | --- | --- | | 113 | 15 | 39 | 5 |   Ideally the app should be able to send all Cycles (1st cycle, 2nd cycle, 3rd cycle, 4th and subsequent cycles) to the System.   1. Notifies the user when the system is ready for treatment |
| Patient begins first Cycle | 1. Waits for commands from App 2. Receives command from the app to begin the first Cycle 3. Starts treatment | 1. Prompts user to enter their pain level and symptom severity 2. Sends pain and symptom information to Thermaquil’s AWS database. 3. Sends start command to begin first Cycle |
| Cycle timer reaches 0\* | 1. Transitions between specified hot/cold cycle 2. Logs event and sends event to app | 1. Retrieves event from device and sends event along with total cycle duration to Thermaquil’s AWS database. |
| Temperature changed using App\* | 1. Receives command from app to adjust temperature. 2. Adjusts temperature set point 3. Logs event and sends event to app | 1. Receives command from user 2. Sends command to device to adjust temperature 3. Receives log from device and sends event log to Thermaquil’s AWS database. |
| Cycle duration changed using App\* | 1. Receives command from app to adjust timer 2. Adjusts time remaining in cycle 3. Records event/log and sends to App 4. Receives updated Cycle information from App, if applicable. | 1. Receives command from user 2. Sends command to device to adjust time remaining in cycle 3. Receives log from device and sends log to Thermaquil’s AWS database 4. If the Cycle duration was increased beyond the planned duration for the next Cycle of that type (hot or cold), update future cycles to use the adjusted duration and send updated cycle information to System |
| Change Cycle button activated using the App | 1. Receives command from the App to change cycles. 2. Transitions between Hot/Cold Cycle 3. Records event/log 4. Sends event/log to the App 5. Starts timer based on new Cycle duration | 1. Receives command from the User 2. Sends command to the System to transition between Hot/Cold Cycle 3. Receives log from device and sends log to database |
| User presses pause button on App\* | 1. Receives command from App 2. Pauses timer and maintains current temperature 3. Fluid flow to the pad stops 4. Waits for command to resume from App 5. Records event/log 6. Sends event/log to the App | 1. Receives command from the User 2. Sends command to the System to pause treatment 3. Informs user that the system is paused and will turn itself off in 30 minutes 4. Receives event/log from the System and sends it to Thermaquil’s AWS database |
| User presses resume button on App | 1. Receives command from App 2. Starts timer from paused time 3. Records event/log 4. Sends event/log to the App | 1. Receives command from the User 2. Sends command to the System to resume treatment 3. Receives event/log from the System and sends it to Thermaquil’s AWS database |
| User ends treatment via App | 1. Receives command from App 2. Records event/log 3. Turns off heater, cooler, pumps | 1. User presses stop treatment button 2. Sends command to the System 3. Receives event/log from System 4. Prompts Patient to enter their pain level and symptom severity 5. Sends pain/symptom info to Thermaquil’s database 6. Prompts user to enter their pain/symptoms 30, 60, 120, and 300 minutes after treatment. |
| User ends treatment via Stop button on System | 1. Records event/log 2. Sends event/log to App 3. Turns off heater, cooler, pumps 4. May receive new cycle information for user profile from App | 1. Receives event/log from System 2. Receives event/log from System 3. May send new cycle information for user profile to System 4. Prompts Patient to enter their pain level and symptom severity 5. Sends pain/symptom info to Thermaquil’s database 6. Prompts user to enter their pain/symptoms 30, 60, 120, and 300 minutes after treatment. |

\*see if/then scenarios for more detail or additional scenarios

## Second and subsequent use

| Step | Device actions | App actions |
| --- | --- | --- |
| The user turns on the system | 1. System powers on 2. System reads a pre-programmed list of cycles that was previously sent by the App 3. System connects to App | 1. Connects to device or displays pairing instructions 2. Requests log from System 3. Notifies patient that connection is successful and they are ready to start treatment |
| Patient is prepared to start treatment | 1. Receives new cycle information from the App or uses previously sent list of cycles to heat/cool reservoirs according to the first Cycle temperatures | 1. App may send new set of cycle information to System. Example cycles:  | Hot temp | Hot time | Cold temp | Cold time | | --- | --- | --- | --- | | 119 | 45 | 30 | 10 | | 119 | 45 | 30 | 30 | |
| Treatment starts | 1. If connected, waits for command from App to start treatment. 2. Receives command from the app to begin the first Cycle 3. Starts treatment | 1. User presses start button 2. App prompts user to enter their pain level and symptom severity 3. Sends pain and symptom information to Thermaquil’s AWS database. 4. Sends start command to begin first Cycle |
| First Cycle completed | 1. Transition to second Cycle, third Cycle, etc. until user changes settings, final Cycle is reached, or System receives command from App to stop or pause treatment. 2. Sends log/event to App | 1. Receives log/even from System. 2. Cycles continue until paused, stopped, or the final Cycle is reached. |
| Final cycle timer reaches 0. | 1. Timer for final planned cold cycle reaches 0. 2. Saves event/log and sends to App 3. Powers down heater, cooler, pumps | 1. Timer for final planned cold cycle reaches 0. 2. Receives event/log from app and sends to Thermaquil’s database 3. If starting pain was 4 or higher, prompts user to enter their pain/symptoms, send pain/symptoms to Thermaquil’s database, etc. |
| Patient ends treatment via App | 1. Receives command from App 2. Turns off heater/cooler/ pumps 3. Saves event/log and sends to App 4. May receive updated treatment program for the patient’s profile. 5. If received, saves updated treatment to patient profile | 1. Sends end treatment command to System 2. Prompts user to enter their pain/symptom info 3. Receives event/log from System 4. Sends event/log and pain/symptom info to Thermaquil’s database |
| Patient ends treatment via stop button | 1. Patient presses stop button on device 2. Turns off heater/cooler/ pumps 3. Saves event/log and sends to App 4. May receive updated treatment program for the patient’s profile. 5. If received, saves updated treatment to patient profile | 1. Receives event/log from System 2. If starting pain was 4 or higher, prompts user to enter their pain/symptoms 3. Sends event/log and pain/symptoms to database |

## Reservoir filling

| Scenario | Device action(s) | App action(s) |
| --- | --- | --- |
| User decides to fill reservoir (one step filling/topping off reservoirs) | 1. receives command from App indicating that the reservoir is being filled 2. sends reservoir status to App every second to allow App to monitor filling process 3. pumps fluid from hot to cold reservoir until System detects fluid in the cold reservoir 4. sends status indicating that filling is complete when fluid is detected in both reservoirs | 1. User uses GUI to select fill reservoir 2. Requests reservoir status from System 3. Shows user instructions to fill reservoir and monitors reservoir status from System    1. While cold reservoir is not full OR cold reservoir is not full, display instructions to slowly add fluid    2. If both reservoirs are full, displays instructions to stop adding fluid and to reattach the cap for the cold reservoir. |

# If/then scenarios

## Base use cases and expected scenarios

| Scenario | Device action(s) | App action(s) |
| --- | --- | --- |
| Short/temporary Bluetooth disconnection during a temperature cycle caused by a random interruption or the app being moved too far from the device. | 1. Device continues to deliver therapy 2. Continues to record log to local storage 3. Continues timer and transitions to the next Cycle if the timer reaches 0. 4. Attempts to reconnect to app 5. When connected, sends logfiles to App | 1. Attempts to reconnect to device 2. Alerts User that the System is not connected 3. Continues timer for current temperature cycle 4. If not reconnected within a few seconds, notifies user that Bluetooth connection is lost and provides instructions for reconnecting 5. When reconnected, retrieves log files from device |
| Long/permanent Bluetooth disconnection during a cycle (e.g., tablet dies) | 1. Device continues to deliver therapy 2. Continues to record log to local storage 3. Continues timer and transitions to the next Cycle if the timer reaches 0. 4. Attempts to reconnect to app | 1. Same workflow as short disconnection if App is running. |
| Cycle timer ends while app is not connected | 1. Time remaining in the Cycle reaches 0 2. Transitions between hot/cold cycle 3. Records event/log and attempts to send information to the app 4. starts timer for next Cycle 5. sends logs to App when reconnected | 1. Alerts user that the device is not connected as soon as disconnection happens 2. Requests logs from system when reconnected 3. Receives log of event from the System and sends log to Thermaquil’s AWS database. |
| User uses power switch to turn off device instead of using the button on the App | 1. Device powers off 2. Logfile and local storage are not corrupted | 1. App recognizes that the device is powered off, if possible 2. Asks User if therapy is finished 3. If therapy is finished, records time of power loss as end of therapy 4. Requests pain/symptom information from the patient. |

## Book ends/edge cases

| Scenario | Device action(s) | App action(s) |
| --- | --- | --- |
| User tries to use the System without the App after already completing their first treatment. | 1. User powers on device and presses start button 2. System reads treatment program from local storage and prepares for therapy 3. System records log to local storage 4. System tries to connect to App 5. When connected to the App, System uploads missing log files | If connected during treatment:   1. Requests and receives logfiles 2. Reads device status and adjusts display based on current treatment. |
| User tries to use the System without the App before completing their first treatment | 1. System does not prepare for therapy or start. 2. System records log to local storage | 1. Is turned off or screen is locked pending correct passcode entry. |
| User pauses therapy and does not resume within 30 minutes | 1. Timer begins when therapy is paused 2. Waits for command from App 3. Receives command from App to resume, receives command to restart timer because the user will resume, or continues to pause without input. 4. After timer reaches 60 minutes, System turns off heater/cooler and pumps to save power. 5. Records event/log 6. Sends event/log to App, if connected 7. Waits for command from the App | 1. After 30 minutes, asks user if they stopped treatment and informs them that the device will power down in 30 minutes without their input. 2. If treatment is finished, records time paused as end time, sends stop treatment command to System, prompts user for pain/symptom info. 3. If therapy is not finished, sends command to System to remain paused instead of turning off heater/cooler/pumps |
| After disconnection, App and System report different Cycle type, target temperature, and/or time remaining in the Cycle. | 1. Receives request and sends log to App 2. Receives cycle info from App and adjusts temperature/time remaining to match App 3. Saves event/log and sends to App | 1. Requests log from System. 2. Sends Cycle information to System 3. Receives event/log from System 4. Sends event/log to database |

# Risk Mitigation Features

| Situation, event, and/or harm | Mitigation |
| --- | --- |
| Runaway heating caused by empty hot reservoir, software error, or other malfunction | Thermal fuse disconnects power to the heater if the heater becomes too hot |
| User falls asleep while using the system | User profile contains default treatment program of fixed duration. System does not continue beyond default program without user input. |
| User selects a temperature/time combination likely to cause burn or to damage the device (e.g. 200C or -200C) | User is only able to increase/decrease temperature within a pre-defined range. The temperature range shall be adjustable by a company representative with appropriate access. |
| User pauses therapy and does not take off their pad, resulting in longer than expected exposure to the temperature | System does not pump fluid to the pad while paused. |
| Temperature probe becomes disconnected or reports unreasonable temperatures | System turns off if temperature readings are unreasonable. Unreasonable readings could include:   * Negative temperatures * Temperature < 15 F * Temperature > 170 F |

# Nice to haves

* Adjustable settings/profiles for different pads
* Resume therapy if the System loses power
* Ability to update firmware via Bluetooth or USB instead of specialized tool
* Automatically top off reservoirs during therapy
* Ability to charge and communicate with a tablet that is plugged into the device