**1. Introduction**

*1.1 Purpose of the system*

The Smart Thermostat/Ventilation System will be open and close vents throughout the house depending on the temperatures of its rooms to allow for optimal temperature control and heating efficiency. As an additional feature, the system will detect for gas leaks, excessive temperatures within the vents, and water leaks near critical systems

*1.2 Scope of the system*

The Smart Thermostat/Ventilation System will be responsible for the following:

The central control system -

The ventilation system - the air ducts, the air direction system which includes the motors and flaps directing the air, the excessive heat sensors within the air ducts

The safety sensor network - the carbon monoxide detector, the moisture detector, and the excessive heat detector, and the small area around them in which they monitor.

The room temperature sensor network - the heat sensors located within each room and the small area in which the detectors monitor.

The furnace - the heat it produces, and if it is activated or not

*1.3 Objectives and success criteria of the project*

To decrease the total energy required to heat a building, heat the building more evenly and to allow for room heat customization, to make a safer heating system by setting up monitoring systems for malfunctions and leaks,

|  |  |
| --- | --- |
| **2. Current system**  Most homes have a basic furnace that sends hot air through an open ventilation system. Some buildings have vent openings that can control the airflow to a room from an open/close wall vent. However, these must be adjusted manually and do not prevent heat from being generated by the furnace and then getting trapped in the air ducts behind the closed vent. Additionally, the furnace will only be activated by a single temperature sensor. This leaves the possibility that other rooms in the house are exposed to much more, or much less heat than the room being monitored. In saying this, a situation such as a user being in a room that is much colder than the monitored room will result in the user adjusting the thermostat so that the room is warmer. In doing this, the furnace would heat the entire house and that room which makes the traditional furnace very inefficient. Lastly, the furnace is considered “blind” to its surroundings. When told to, the furnace will attempt to run under any condition, be it a foot of water encompassing the furnace, a gas leak that will cause an explosion if there is an ignition source, or if the vents are so full of hot air that any more would endanger the building. |  |

|  |
| --- |
| **3. Proposed system** |
| Eric and I plan to solve each of the issues listed above. Firstly, we would install a central control system that would allow the Furnace, ventilation system controls, and monitoring devices to communicate. Temperature sensors would be installed within each room, they would communicate to the central control system their temperature measurements. We would install a system within the ventilation ducts that would open and close ventilation pathways based upon which rooms require and do not require heat provided to them. This would reduce the amount of hot air sitting within the vents and allow the system to be more efficient in heating the rooms that require heat, and leaving the already warm rooms alone. Lastly, monitoring systems such as a carbon monoxide detector, moisture detector, and an excessive heat detector would be placed around the furnace and ventilation system. These would communicate if a problem arises with the central control system and as a result, would shut down the furnace. |
| *3.1 Overview* |
| The system will be composed of 5 main components, the central control system, the ventilation system that directs air flow, the safety sensor network, the room temperature sensor network, and the furnace. The system will all connect to the central control system. The safety and room temperature control sensors will only send their readings to the control system. The furnace and ventilation system will receive instructions from the central control system and return either success or failure on said instructions. In depth, the ventilation system will be a series of motors attached to flaps that will be engaged to block airflow and direct it through other channels. |
| *3.2 Functional requirements* |
| • The central control system must run on an architecture that will allow for communication with a database, voltage readings from an input terminal, and display a command line.  • The ventilation system must have a strong enough motor system such that a single “node” of motors can lift the air flow directional pane. Additionally, the ventilation system must be able to detect the orientation of of the air flow directional panes and return an on/off status.  • The safety sensor network must be able to send an on/off signal to the central control system without outside request or interaction  • The temperature sensor network must be able to communicate the temperature of the room via a certain voltage. The communication must be at least every 5 minutes and the device should communicate without being prompted by the central control system  • The furnace must be able to receive an on/off command, and reply it’s on/off status. |
| *3.3 Non-functional requirements*  *3.3.1 Usability*  The system must be fully accessible to the admin and must restrict some functionalities based upon which user that is using the system. The system must be able to be installed in all buildings that contain a single heating source (furnace).  *3.3.2 Reliability*  The system must be able to function on its own and without direct human intervention of long periods of time. The system must also be able to be turned off instantly in the event of a problem occurring outside of the system’s scope. In the event of a failure by the system, there must be a method in place to inform the users of the issue and when it occurred.  *3.3.3 Performance*  The system must at least match the efficiency of the traditional furnace system. The system must heat rooms within a reasonable speed, and be able to react to changes made to the system within a short period of time.  *3.3.4 Supportability*  The system must be built modularly so that as time progresses the team can implement extensions to the system without having to take the entire system into account. Additionally, with a modular system, extensions can be added and removed without impacting the system at large.  *3.3.5 Implementation*  N/A  *3.3.6 Interface*  The interface must be easily navigated for users with a less technical background. The structure will be a deep hierarchy with descriptive tabs so that users can do what they want to do without feeling overwhelmed. |
|  |
| **3.4 System models**  *3.4.1 Scenarios*  *3.4.1.1 User Stories*  As a(n)...  • convenience-­seeking user  • efficiency-­seeking user  • elderly user  • child  • pet  • administrator  • comfort­-seeking  • user who's out of the house often  • user with sensitive objects like plants or antiques  I would like to automate the heat distribution in my house...  • so I don't have to do it  • so the temperature's always comfortable  • because I'm incapable of changing the temperature myself  • so I waste less energy  • so I can be notified if there's a problem (with special detectors)  • so the temperature remains constant  I would like the have a system that monitors my furnace and its safety …  • so that my house is safer  • so I and my family can be safer  • so that I can leave on trips and not have to worry  • so I don’t have to do regular checks on my system  • so I can let my kids change the system  • so that I can be alerted to other problems in my house  *3.4.1.2 User Scenarios*  Name: Alter state of gate (open or close)  Description: Opens a closed gate or closes an open gate depending on the temperature of its corresponding room(s).  Actors: user, UI, gate, thermometer, system controller  Basic flow:  1. Controller sends requests for temperature  2. Thermometer reports temperature  3. Controller sends appropriate signal to gate  4. Gate responds:  • in this case, temperature is at an unbalanced level (too hot or too cold)  • if room is too hot, close  • if room is too cold, open  Alternate flow:  1. User notifies UI about desired change  2. UI sends signal to controller  3. Controller sends appropriate signal to gate  4. Gate responds:  • if user requires heat, open  • if user does not require heat, close  Preconditions:  • Room cannot be in an 'invalid state' (when the room is above temperature and the gate is closed or vice versa). If this occurs, gate will not respond to signal (since the issue must be external).  Postconditions:  • gate returns signal to controller:  • 0 if operation was successful  • 0 if no change (see precondition)  • 1 if unsuccessful (door is jammed, etc.)  Name: Report temperatures  Description: Thermometer reports the temperatures to the system controller.  Actors: thermometer, system controller, user, UI  Basic flow:  1. System controller sends periodic request for temperature  2. Thermometer sends temperature to system controller  Alternate flow:  1. User requests to view temperatures  2. System controller sends request for temperature from thermometer  3. Thermometer sends temperature to system controller  Preconditions:  • Thermometer is calibrated and working  Postconditions:  • Value is sent to system controller (in this case, the temperature)  Name: Display layout  Description: Displays the house layout to the user via the UI. Also shows orientation of gates and temperatures of rooms.  Actors: user, UI, gate, thermometer, system controller  Basic flow:  1. User requests to view layout of house via the UI  2. UI sends request to system controller  3. Controller checks status of gates  4. Controller checks temperatures of rooms  5. Controller sends data back to UI  6. UI displays information for user in form of a house schematic  Alternate flow:  • None, will only take place when user requests to view layout  Preconditions:  • Thermometers are calibrated and working  Postconditions:  N/A  Name: Display warnings  Description: Displays any critical warnings to the user via the UI.  Actors: user, UI, system controller  Basic flow:  1. System controller performs scheduled checkup  2. Controller receives an error code (Gate failed to change state, detector reports anomaly)  3. Controller sends error code to UI  4. UI displays error code  Alternate flow:  1. Detector detects critical state (excessive heat, gas leak, water flood)  2. Detector sends immediate signal to system controller  3. Controller sends error code to UI  4. UI produces sound via thermostat to alert user immediately  Preconditions:  • Thermometers are calibrated and working  • Detectors are functioning properly  Postconditions:  • N/A |
| Name: Furnace  Description: Provide heat for the house when turned on by the system controller  Actors:  Basic Flow:  1.System Controller tells Furnace to turn on  2.Furnace turns on  3.Furnace produces heats that is sent to the ventilation system  4.System controller tells Furnace to turn off  5.Furnace turns off  Alternate Flow:  N/A  Preconditions:  • Furnace is turned off  • Furnace is connected to the system controller which guides it.  Postconditions:  • Furnace is turned off  • Furnace provided heat to the ventilation system    Name: Ventilation heat detector  Description: a device that detects the internal heat of the ventilation system  Actors:  Basic Flow: Heat detector detects excessive heat levels -> heat detector sends warning signal to system controller  Alternate Flow:  Heat detector detects normal heat levels -> heat detector idles  Preconditions:  • Heat detector is located within the ventilation system  Post conditions:  N/A    Name: Carbon monoxide detector  Description:  Actors:  Basic Flow: Carbon monoxide detector detects excessive levels of CO -> Carbon monoxide detector sends warning signal to system controller  Alternate Flow:  Carbon monoxide detector detects normal CO levels  Carbon monoxide detector idles  Preconditions:  • Carbon Monoxide is located near gas sources  Postconditions:  N/A  Name: Moisture detector  Description: a device that detects the moisture around the furnace  Actors:  Basic Flow:   1. Moisture detector detects moisture around the base of the furnace 2. Moisture detector sends warning signal to system controller   Alternate Flow:   1. Moisture detector detects normal heat levels 2. Moisture detector idles   Preconditions:  • Moisture detector is located within the ventilation system  Postconditions:  N/A    Name: Room temperature sensors  Description: a device that measures the room temperature of a room and returns the measurement as a certain voltage.  Actors:  Basic Flow:   1. Temperature detector measures the temperature of the room 2. Temperature detector sends the temperature of the room to the central control unit   Alternate Flow:   1. Carbon monoxide detector detects normal CO levels 2. Carbon monoxide detector idles   Preconditions:  • Carbon Monoxide is located near gas sources  Postconditions:  N/A  3.4.2 Use case model  furnace_usecasediagram2.png  3.4.3 Object model  N/A  3.4.4 Dynamic model  N/A  3.4.5 User interface—navigational paths and screen mock-ups  Please refer to the attached document or this URL: <https://onedrive.live.com/edit.aspx/Documents/ENSE%20374?cid=c2fa46625bba5787&id=documents&wd=target%28Homework.one%7C364383C9-5AEF-432C-83E5-8EA5AC7FA896%2FUser%20Interface%7C47B1A732-C0C8-4B03-B11C-40D4169D5C29%2F%29> |
| **4. Glossary** |
| N/A |