Icehouse Challenge System Proposal: DeSIGN (NMSU Team)

1. Introduction

We develop a wearable, context-aware system to support disaster responders. We use Sony Smart Eyeglasses to visualize team status, recommend context-sensitive action, and enable team communication via voice-to-text. We use sensor data (i.e., game state) provided by the EReps to capture Icehouse world context and player state, aiding decision making. The Eyeglasses' input controller provides minimal means of interaction. We have designed the system with real-world contexts in mind, but will build it for the present game.

To account for the game rules, we have created a hierarchy of activities (e.g., hazard response, victim assistance, etc.), and recommendations to the player are made accordingly (e.g, victims, fires, traps are prioritized first). The system is designed not to let critical information be missed (e.g., critical exposure level). To make sure the UI is useful and effective with the chosen hardware, we work from guidelines provided by Sony and known best practices.

2. Interaction

The system requires minimal physical interaction from the user, whom we expect to be physically active in the environment, using hands for non-information tasks, and/or unable to manipulate a device due to personal protective equipment. The present design provides three display modes: map, command, and worker. The map display is shown when the user raises a hand in front of her/his display. The other two views are toggled using the controller's Back button. We expect non-command workers to use the worker view, and the commander to use the command view, but each player type may find the other interface useful. Finally, orders and/or status updates can be relayed by triggering the Talk button, which enables a worker to dictate a text message to all workers or a specific one using worker and/or room identifiers (see next section).

Communication among team members will be done through audio/text messaging. The voice-to-text functionality of the glasses will be used to convert audio to text and players will have the option to broadcast messages. Critical alerts will also be shown in a timely manner.

3. Identifiers

A key concept in the present design is *identifiers*. Workers and locations are named by single-letter identifiers that are spoken using the NATO phonetic alphabet (which is familiar to responders). During the loadout phase, each player is assigned an identifier (e.g., "Alpha", "Bravo", "Charlie", or "Commander").

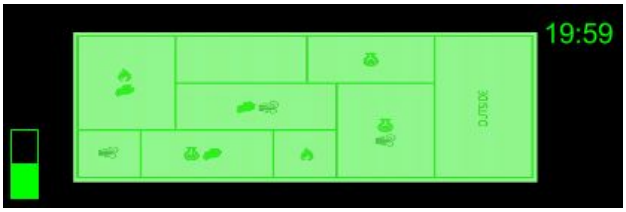
Each room in the Icehouse world is assigned a unique identifier, starting from the tail of the alphabet and named according to the order they are explored. The first room encountered will be "Zulu", the second will be "Yankee", etc. Room identifiers are not connected to the actual map (which is beyond the scope of this project) but contribute to a room graph that can be used for simple navigation (see section 7).

4. Context-Sensitive Display Modes

The primary purpose of the designed system is to provide players with a clear sense of the state of the environment and of team needs. The display has three modes: map, command, and worker.

In all modes, a worker is provided with basic information. A bar graph at the bottom-left shows exposure level, while total remaining game time is in the top-right corner. A hollow bar indicates a worker has no exposure. The bar fills as the exposure level increases (garnering more visual attention). As the worker enters a room, an alert in the center of the screen guides him/her about his tasks (see section 5).

4.1. Map Mode



In map mode, the system displays a monochrome map of the Icehouse world. The map is digitized during the loadout phase, using a camera. The map is static, and is used only to help players do wayfinding. Map mode is activated by the user holding her/his hand in front of her/his face.

4.2. Command Mode

Command mode is primarily for the commander. It provides view that aids in situation awareness, showing the state of each team member in a compact representation at the bottom of the screen. For each player, the commander can see the player's identifier, skill icon, exposure level, room identifier, and activity status represented by an icon for the activity the worker is undertaking and a countdown timer for the estimated completion time. Workers can toggle between command and worker modes by pressing the "Back" button.

From the figure above, the Commander has the damage control skill, no exposure, and is handling a fire hazard in room Z. Alpha is a hazmat specialist, has received full exposure, is in room Y, and is incapacitated. Bravo is a medic and will be busy for 1 minute, 20 seconds handling a liquid hazard. Charlie has died.

This view shows the commander which worker will be free the earliest and can use him/her for another task, or, alternatively, which worker needs emergency aid. For example, if he sees that a worker is working on high-priority hazard (e.g fire), then s/he might re-task someone working on a low-priority hazard (e.g., a liquid hazard).

4.3. Worker Mode

Worker mode (figure to right) provides an player with details of her/his state in the Icehouse world. As the worker enters a room, an alert indicates the highest priority task.

After a worker taps the phone, the view shows hazards present in a that room, along with their class (denoted by vertical bars). The system shows a suggestion by highlighting a hazard with a box. In the figure, the system suggests that the worker should try to eliminate class 2 Fire first. It is up to the worker to decide which hazard to work on. This view also shows victims with their priority. If the victims are found and their action card is tapped, the system shows victims with their priority and recommends which victim should be treated first. In the figure, system suggests treating victim with priority 6 first.

5. Heuristic Activity Priority Engine

Activity priority is based on the hierarchy and class of hazards. The system prioritizes triaging victims. In the case of hazards, it prioritizes Traps, Smoke, Fire, Gas and then Liquid. We will develop (and code) a detailed hierarchy based on the game rules.

6. Activity Time Estimation

The activity time of the worker is calculated automatically, accounting for shared work on the same task. It is calculated using the Icehouse game rules and the state of all players in the game. For example, if a worker was working on a class 2 fire and has already gained 1 progress points, and another worker joins him then the activity time of both the workers will change from 5:00 minutes to 2:30 minutes.

7. Dynamic Room Adjacency Graph Construction

We will implement an algorithm to dynamically build an adjacency graph of the environment as workers discover rooms. The graph identifies which rooms are connected to one another, and can be used to define the shortest path (in terms of rooms, not physical distance). Room identifiers start with “Z” for the first room entered followed by “Y”, “X” so on. The adjacency graph will be populated using workers’ “tap ins” (we assume that consecutive rooms are adjacent and accessible). This enables the system to provide a worker with a directed graph for navigation when s/he is sent to a particular location. For example, if the commander tells worker Alpha to assist Bravo, the system will automatically show the user the sequence of rooms to get there, based on the data that has been collected so far (e.g., W→Y→X).

8. Loadout

After a physical map is obtained, we will take a picture of the map which can be accessed by the workers through the user interface. The selection of the tech items will be done based on the map and the number and types of hazard present. We will have at least one medic, and damage control to deal with victims, fire and traps, the other two will be picked based on the map. The loadout will start with a 2-3 minute video of the system, prepared in advance, which will be shown to the workers to guide them on using the system.

9. Demonstration of Interface