

Sprint: 3

From: 05/06/2024 – 05/17/2024

Team: ERA: Emergency Response Assist

Team Member	Tickets	Points
Jatin Madan	3	12
Vaishnavi Sunil Desai	3	12
Isha Ghiria	3	12
Sharvesh Patki	2	12

Sprint Overview:

Planned		Completed	
Items	Points	Items	Points
11	48	11	48

Sprint Retrospective:

- What have you done during this sprint?
 - Jatin Madan
 - Jatin worked on exposing an API on the Main ERA Server to poll data from the Gunshot Detection API, which has the capabilities to stream continuous audio as well as scan uploaded media files.

```

    curl -X 'POST' \
      http://0.0.0.0:8000/upload/ \
      -H 'accept: application/json' \
      -H 'Content-Type: multipart/form-data' \
      -F 'audio_file=@99180-9-0-49.wav;type=audio/wav'
  
```

test URL
http://0.0.0.0:8000/upload/
or response

Details

Response body

```

  {
    "Detected": "Street Music"
  }
  
```

Download

Response headers

```

  content-length: 27
  content-type: application/json
  date: Thu, 02 May 2024 18:42:14 GMT
  server: uvicorn
  
```

Responses

Description

Successful Response

Media type

No links

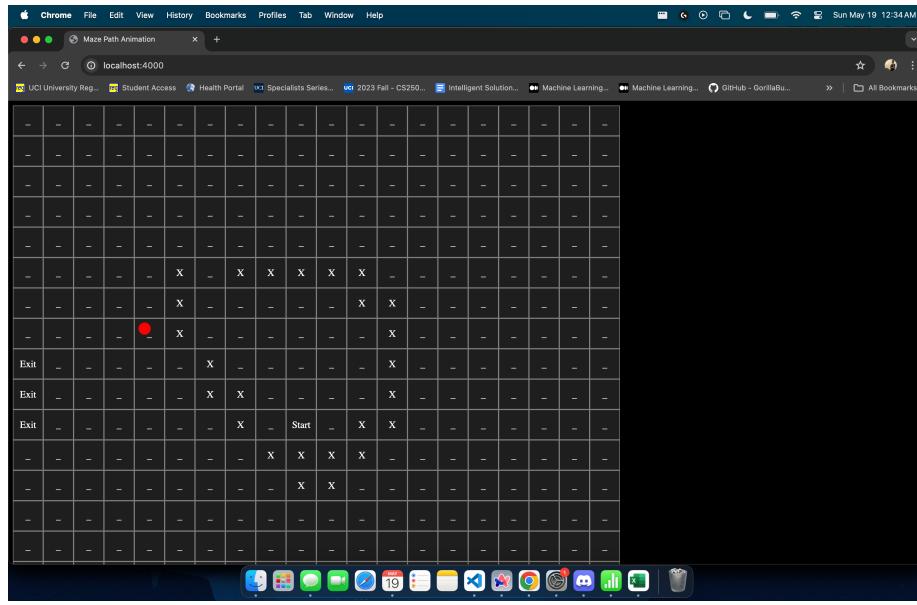
- Jatin also worked on modifying the Main ERA Server to include capabilities to initiate Triggers for the Gunshot Machine Learning API.

		CLASS	ACCURACY
6	Gun Shot	99.248120	
7	Jackhammer	98.592540	
5	Engine Idling	97.226174	
4	Drilling	96.169916	
8	Siren	96.126482	
1	Car Horn	95.573770	
0	Air Conditioner	94.608960	
3	Dog bark	93.807829	
9	Street Music	92.512246	
accuracy		0.95	12189
macro avg	0.95	0.95	12189
weighted avg	0.95	0.95	12189
2	Children Playing	91.739766	

- Jatin also worked on developing an algorithm that would be able to predict and triangulate a user's approximate location in a specified radius.

- Isha Ghiria

- Isha worked on updating the A* algorithm used for escape route planning to include multiple exit points during the search



- Isha also worked on transforming the Floor Plan Data into Escape Route Algorithm Readable Format. This serves as the basis for the rest of the application to consume and analyze.

```

class Floorplan {
  public deletePlanName: any, callback: (err: any) => void {
    // Delete plan from database by name
    const deleteQuery = `DELETE FROM plans WHERE name = ?`;
    db.run(deleteQuery, [name], function (err) {
      if (err) {
        callback(err);
      } else {
        callback(null);
      }
    });
  }

  public transformMaze(plan: PlanRow): number[][] {
    let rows: number = plan.height;
    let cols: number = plan.width;
    var maze: number[][] = new Array(rows);

    for (let i = 0; i < rows; i++) {
      maze[i] = new Array(cols).fill(0);
    }

    // Blocker / walls
    const data = JSON.parse(plan.data.toString());
    data.blocked.forEach((point: { x: number; y: number }) => {
      maze[point.x][point.y] = 1;
    });

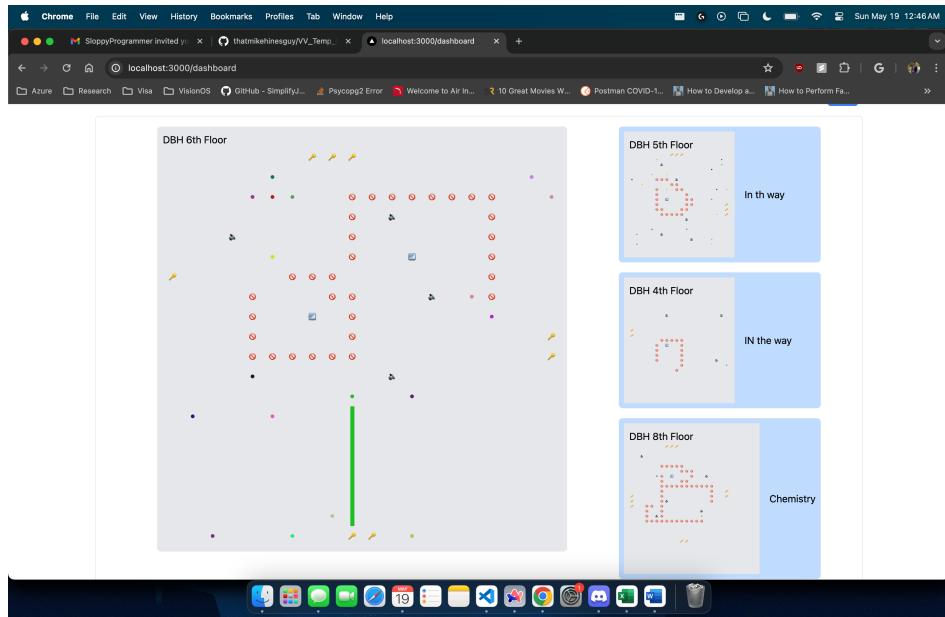
    // Access points
    plan.data.access.forEach((point: { x: number; y: number }) => {
      // maze[point.x][point.y] = 9;
    });

    return maze;
  }

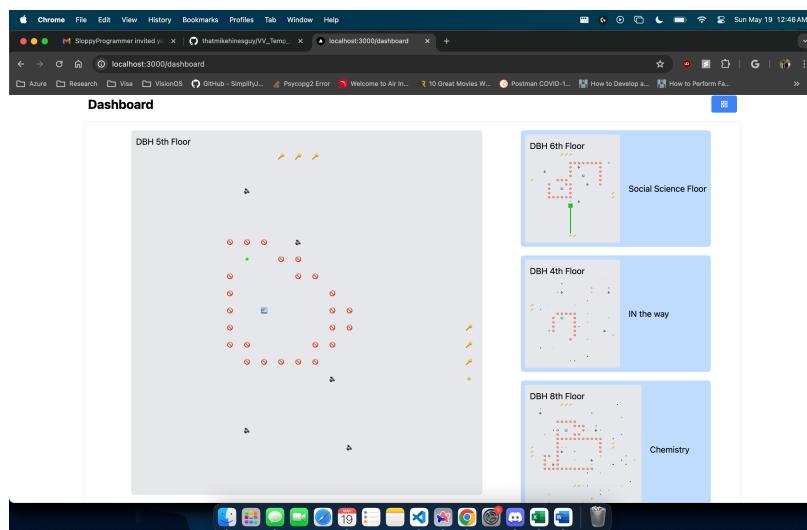
  public getAccessPoints(plan: PlanRow): myNode[] {
    var accessPoints: myNode[] = [];
    const data = JSON.parse(plan.data.toString());
    data.access.forEach((api: { x: number; y: number }) => {
      const node = new myNode(api.x, api.y);
      accessPoints.push(node);
    });
  }
}

```

- Isha also worked on exposing an API on the Main ERA Server to plan the shortest route to the closest exit point for each user.



- Vaishnavi Sunil Desai
 - Vaishnavi worked on materializing and updating the UI Component of Escape Route Plan



- Vaishnavi also worked on exposing an API on the Main ERA Server to poll data from Network Logs Simulator

```

class LocationProcessorCtrl {
    /**
     * Returns void
     */
    public static setDb(db: any): void {
        router.route('/user/location')
            .get((req: express.Request, res: express.Response) => {
                getValidator.validate(req, LocationProcessorCtrl.getterLocation);
            })
    }

    /**
     * The method getStates. undefined
     */
    @param req of type express.Request
    @param res of type express.Response
    @returns void
    */
    public static getLocation(req: express.Request, res: express.Response): void {
        console.log('getters location -', req.url);
        var floorName: string = req.query.name ? req.query.name.toString() : 'OBW286tW2#flor';
        UserLocation.getLocation(floorName).then(floorPlanData => {
            console.log('calling');
            res.writeHead(200, {
                'Content-Type': 'text/plain',
                'Transfer-Encoding': 'chunked'
            });
            UserLocation.getUserLocation(floorName, LocationProcessorCtrl.processUserLocation, res, floorPlanData.plan).then(res => {
                res.end();
            }).catch(err => {
                res.write(JSON.stringify(err));
                res.end();
            });
        });
    }

    public static processUserLocation(parsed: any, res: express.Response, floorPlanData: any) {
        // consider parsed.floorPlanChunks[0].parsed;
        var userLocation = parsed.location;
        userLocation.getAssociatedAccessPoint(parsed, floorPlanData);
        res.write(JSON.stringify(userLocation));
        res.end();
    }
}

```

- Vaishnavi also worked on creating a triangulation algorithm using WIFI Access Point Logs

```

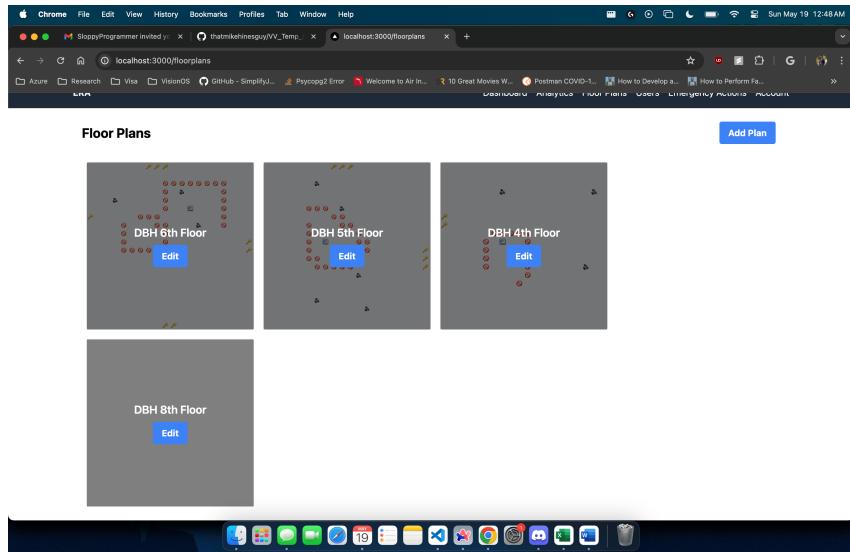
public static processUserLocation(chunk: any, floorPlan: any): {MAC Address": string, location: {x: number, y: number}} {
    var position = this.floorPlan.parseChunk(chunk);
    var radius = this.calculateSignalStrength(chunk['RSSI']);
    var randomLocation = this.pickRandomPoint(this.generatePointInCircle(position.x, position.y, radius, floorPlan.width, floorPlan.height));
    return {MAC Address": chunk["MAC Address"], location: randomLocation};
}

private static calculateSignalStrength(signalStrength: number): number {
    // Assuming a simple signal strength model where signal strength decreases linearly with distance
    const maxSignalStrength = 100; // Maximum signal strength
    const maxDistance = 100; // Maximum distance for full signal strength
    const distance = (maxDistance + (maxSignalStrength - signalStrength)) / maxSignalStrength;
    return distance;
}

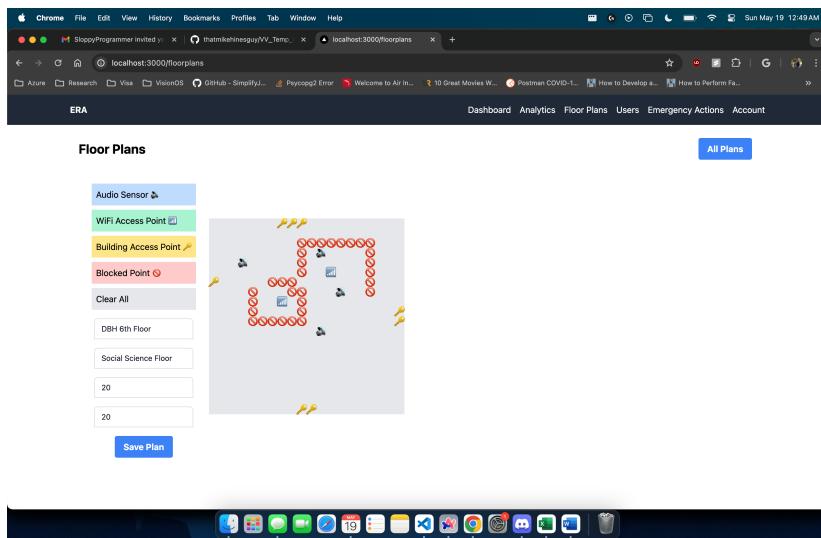
private static generatePointInCircle(x: number, y: number, radius: number, width: number, height: number): {x: number, y: number} {
    const points: {x: number, y: number}[] = [];
    for (let theta = 0; theta < 2 * Math.PI; theta += (1 / radius)) {
        const x2 = x + radius * Math.cos(theta);
        const y2 = y + radius * Math.sin(theta);
        if ((x2 >= 0 && x2 <= width) && (y2 >= 0 && y2 <= height)) {
            if (!points.length === 0) {
                points.push({x: x2, y: y2});
            } else {
                points.push({x: x2, y: y2});
            }
        }
    }
    return points;
}

```

- Sharvesh Patki
 - Sharvesh worked on materializing and updating the UI Component of ERA Administrator Dashboard



- Sharvesh also worked on materializing and updating the UI Component of ERA User Input Module



- What went well?
 - Despite certain issues in task estimation, the team was successfully able to complete all the user stories for this sprint and make considerable progress on building the backbone of the ERA system.
 - The team demonstrated adaptability in responding to changes and adjusting plans as needed to address emerging issues or accommodate new requirements. This flexibility allowed them to maintain progress and keep the sprint on track.
 - The sprint provided opportunities for learning and growth, both individually and as a team. Challenges encountered during the sprint served as valuable learning experiences, helping the team identify areas for improvement and develop new skills.

- The team applied best practices in agile development, such as conducting regular stand-up meetings, holding retrospectives, and using agile tools effectively. These practices contributed to the overall success of the sprint.
- What didn't go well?
 - Team
 - The Task estimation for User Story 8 (Materialize and Update the UI Component of ERA Administrator Dashboard) turned out to be inaccurate, as we went into development for the module, and we had to update our story points to include the updated effort.
- What could/should be improved during the next sprint?
 - We can divide our user stories into smaller atomic tasks, instead of grouping multiple tasks in one umbrella user story.
 - As we faced issue with one of the tasks in Task Estimation, we would be working on improving our estimates for the next sprint as they are crucial for better planning and execution. We would be involving the whole team during our sprint goal planning to have inputs on the task estimates.

Sprint Backlog

ID	Type	Owner	Summary	Status	Estimate
1	User Story	Jatin Madan	Calculate Triangulation Radius based on the Signal Strength	Completed	4
2	User Story	Vaishnavi Desai	Perform User Triangulation using WIFI Access Point Logs	Completed	4
3	User Story	Isha Ghiria	Expose API on the Main Server to retrieve shortest path for each user	Completed	4
4	User Story	Isha Ghiria	Update A* algorithm to take multiple end points	Completed	4
5	User Story	Isha Ghiria	Transform the Floor Plan Data into Escape Route Algorithm Readable Format	Completed	4
6	User Story	Vaishnavi Desai	Expose API on the Main Server to Poll data from Network Logs Simulator	Completed	4
7	User Story	Jatin Madan	Expose API on the Main Server to Poll data from Gunshot Detection API	Completed	4
8	User Story	Sharvesh Patki	Materialize and Update the UI Component of ERA Administrator Dashboard	Completed	6
9	User Story	Vaishnavi Desai	Materialize and Update the UI Component of Escape Route Plan	Completed	4
10	User Story	Sharvesh Patki	Materialize and Update the UI Component of User Input Module	Completed	6
11	User Story	Jatin Madan	Integrate Gunshot Detection API with the Main server to initiate Triggers	Completed	4