This is an inventory management system website for a public airplane company.

Admin

* can buy items, like number of planes, engines, tires, fuel for inventory
* they can see what manager needs by looking at their request forms.
* They can see any planes in inventory and it’s condition.
* They can add or remove employees. Gives them username and password.

Manager

* can see the request and report made by the staff
* They can assign any staff to work on planes for maintenance
* They can see any planes in inventory and it’s condition.
* They can request items like a number of engine, fuel, tire, and planes from manager

Staff

* see what they need to work on.
* They can report on plan’s condition.
* They can see any planes in inventory and it’s condition.

SQL Database

Admin: userID, password

Employee: userID, password, firstname, lastname, employee type, status.

Purchase: Manager ID, plane, fuel, tire, engine

Plane: Name ID, condition, fuel, tire, engine

Storage: plane, fuel, tire, engine

Task: Task ID, target Plane, Fuel, tire 1, tire 2, tire 3, tire 4, tire 5, tire 6, engine, condition, task status, staff involved in this task

Admin

* can approve manager’s purchase and add or delete employees.
* They can update storage table by approving the purchase row.
* They can delete purchase row by either approving or refusing.
  + Approving deletes the purchase row, but updates the storage table
  + Refusing just deletes the purchase row
* They fill out the form to create employee and delete the employee.

Manager

* Can approve or refuse tasks created by staff.
  + Pressing approve button automatically assigns it to the available staff.
* Buy components for plane maintenance.

Staff:

* They can create task by completing a **Plane Inspection Report Form**.
  + It goes to manager’s dashboard and waits for them to approve it.
* They can complete the task that is assigned to them.
  + Pressing complete button will update the storage table and update condition of the plane.

By default, there are 22 planes. 12 is active, 4 of them is out of fuel. 7 of them has tire issue, 3 of them has engine failures, and 1 plane broke down.

Components are tire, engine, fuel, and airplane

Admin’s dashboard. He can see pie chart of all plan’s condition, he can manager’s the request, and staff member, he can buy stuff for the repairmen’s.

Manager functionality update:

Manager can see the conditions of all planes like admins

Manager can assign staff to tasks.

On the task tab, there are “on hold tasks”, “assigned tasks” and “completed tasks”

Staff

* see what they need to work on and what task they have completed.
* They can report on plan’s condition.
* They can see any planes in inventory and it’s condition.
* There are 2 types of staff:
  + Inspector
  + Maintenance

Home:

Card 1: Pie Chart

Card 2: Task = Available or not available

Card 3: Task Completed.

Plane:

Search planes

Report:

Fill out a form and submit it

Tasks:

See the details and click “complete” button. Also, clicking on this button changes the component amounts.

Bootstrap Usage 3/8/2025

Man, writing plain html is possible, but very time consuming. My guts are going to vomit if I start writing html. My software developer pride is going to be hurt.

So, I will use bootstrap as much as possible.

For side bars: [https://getbootstrap.com/docs/5.3/examples/sidebars/#](https://getbootstrap.com/docs/5.3/examples/sidebars/)

Login page: <https://getbootstrap.com/docs/5.3/examples/sign-in/>

Containers/cards/box/item: <https://getbootstrap.com/docs/5.3/examples/album/>

3/22/2025

I am gonna focus plane, storehouse and user tables right now.

Purchase will be next

Task will be last table to work with. Its complex.

3/24/25

Task table is gonna show number of workers needed

When manager is altering the task,

Manager task.php Approve button pressed:

* Task status changes to approve
* Add rows to task staff based on the number of needed workers.
* Changes status of staff from available to busy

Manager tasks reject button:

* Task status changes to reject

My next task is to get the values from html to js, and display it in console.log();

Oooooooo

To display task based on status, I am gonna use post method and refresh it each time.

3/26/25

I finished with task creation and manager changing status.

Now, it’s time to show the task to each staff member

Then the task is completed once all staff member finished the task.

1. I need to display the task to each staff

3/27/2025

SELECT TaskID FROM task WHERE taskStatus = ‘completed’;

From the table taskstaff, select all staffUserID where they have the same taskID

Those staffUserID are foreign keys linked to userID in the table users. I need to change the status to available.

SELECT u.firstname, u.lastname

FROM taskStaff ts

JOIN users u ON ts.staffUserID = u.userID

WHERE ts.taskID = " . $row['TaskID']);

3/31/2025

Now we just implement SQL injection prevention

1. Show input validation by putting wrong username.
2. Show database and username/password
3. Login to admin
4. Explain the dashboard
5. Showcase plane search functionalities
6. Showcase approve purchase functionalities and info bar.
7. Create employees, showcase db, and delete employee. Create new employee and use it
8. Show search employee functionalities
9. Login with manager and explain dashboard