**B-Tech Project Interim Report**

**Sanjay – The Air Mission Planner**

**INDEX**

1. **Abstract [Problem Statement]**
2. **Modules**
3. Login / New User
4. Radar Data
5. Integrate Maps
6. Overlay radar data over maps
7. Navigation to a desired location
8. Searching for a desired location
9. **Use Case – 2**
10. **User Interface**
11. **Technical Challenges Breakdown**
12. **Software and API’s**
13. **Help Resources**
14. **Final Deliverables**
15. **Next Version Plans**
16. **Abstract [Problem Statement]**

If Indian Air Force wants to do some air operation in a certain area, they need to know the status of currently flying commercial / government and private planes/helicopters in that area.   
They receive the status (longitude/latitude/altitude) of the vehicles from radars. But radars give data in analog format i.e. the data is in a tabular format which leaves a scope for improvements in the current system.

The Project Sanjay builds a system which can help

* Interactively visualize the status of air borne vehicles in a particular area.

Further Additions to the project may be

* Suggest optimal traffic free path to reach desired location from a given location.
* Simulate the damage caused to the neighbouring civilization (towns, buildings, vegetation) based on ballistic power of the missiles and bombs dropped off the planes.
* Strategize operations in an area with simulations by computer attack using own planes.

1. **Modules**

The complete problem statement has been broken down into some independent modules which can be later integrated together to make the complete system. Each module has its own constraints and specific functionalities.

1. **Login/New User**

The system is to be deployed on the local server of the organization and shouldn’t be accessible from outside or to any other unauthorized personal.

Hence we need strict security constraints. We need to check against

* Computer Scripts which trying to make forced login.
* Unauthorized Users
* Accessibility from outside the organization.

**New User**

The details to be fetched from new user are:

Name

Phone

Designation

Email

Purpose of Use

Captcha (to check if human)

Once user has entered the information, a mail with this information will be sent to the administrator of the software. He will be shown a confirmation screen that he will soon receive login details after validation of his personal details.

The administrator will check the information for authenticity. If valid, then he will grant access to the user by sending him his login details (username, password) in a mail at the email address provided by him.

**Login**

The login screen should ask for this information from the user.

Name

Password

Captcha (to check if human) – Should appear only if user enters password wrong.

Forget Password?

A password reset link should be send to the email address provided by him.

**Admin:**

The admin can add new users to the system. He receives notification if a new user has requested a account and he creates his login details if the user is authentic and sends details to his email.

1. **Radar Data**

The system will be integrated with the radars later. Currently we use txt files to simulate the radar data. We use a separate file for each radar which contains information regarding the latitude/longitude/altitude of the planes in their region.

**Admin:** Can add new radars to the system. Once these new radars have been added to the system the planes in the vicinity of these radars will start showing up too to very user currently viewing the system.

1. **Integrate Maps**

Our system will display precise geographical information and we fetch this information from google earth. The data from google earth allows us to visually see the buildings, terrain, vegetation and everything which makes the experience all the more immersive.

1. **Overlay radar data over maps**

The planes/helicopters would be simulated moving over the geographical data at the exact location as in the real world (except the delay caused by data transmission and processing). Appropriate 3D models will be overlaid over the map data to provide the exact feel of the system.

1. **Navigation to a desired location**

The user can navigate to the exact location sought by him using the zooming and panning controls which will be provided in the system.

1. **Searching for a desired location**

The system will allow the user to input a location in words and it will be matched automatically to the latitude/longitude of the place and the system would navigate to the exact location.

1. **Use Cases**

**Use Case 1:**

User enters the system for first time  
|

Registers for a new account

|

Receives his login details in email

|

Login into the system with his login details

|

Checks the current status of flights over an area

|

Navigates to a certain area he wants to focus on

|

Searches for a place where he wants to see the flights status

|

Logs off

**Use Case 2:**

Admin logins

|

Checks if some new account requests

|

Checks the current status of flights over an area

|

Navigates to a certain area he wants to focus on

|

Searches for a place where he wants to see the flights status

|

Logs off

1. **User Interface**
2. **Technical Challenges**
3. Implement the login system.
4. Make the login secure and accessible only in the local server and not from internet.
5. Integrate the google earth API into the system.
6. Load and display the 3D model into the system at the location indicated by the radar.
7. Dynamically change the location/path/heading of the models according to the path followed by the real vehicles.
8. Enable searching and navigations of the locations by entering their names or hints.
9. **Softwares & API’s**
10. Google Earth API

<http://code.google.com/apis/earth/>

1. 3ds Max
2. Google Warehouse

<http://sketchup.google.com/3dwarehouse/>

1. Unity3D Game Engine

<http://www.unity3d.com>

1. JavaScript
2. **Help Resources**
3. Tutorials and examples given on

[**http://code.google.com/apis/earth/documentation/index.html**](http://code.google.com/apis/earth/documentation/index.html)

1. Tutorials and examples given on

[**http://www.unity3d.com**](http://www.unity3d.com)

1. **Final deliverables**

The final system will have this deliverable.

. A function System ready to be deployed on local server.

1. **Next version Plans**
2. Suggest optimal traffic free path to reach desired location from a given location.
3. Simulate the damage caused to the neighbouring civilization (towns, buildings, vegetation) based on ballistic power of the missiles and bombs dropped off the planes.
4. Strategize operations in an area with simulations by computer attack using own planes.

**\*\*\***