**CSE 240 Project Plan – Deliverable 1**

Deliverable 1 – March 22nd

* Create a class called Date\_Time. It must include members, for minutes, hours, day, month, and year.
* Create a class called HubNode that implements a linked list node. (Please refer to the class diagram for additional information.)
* Create a class called FlightNode that implements a linked list node. (Please refer to the class diagram for additional information.)
* A debugging function that loops over each hub and prints the outgoing flights.
* All dynamic memory allocations must have the corresponding deallocation.
* Reminder: Program must be built using C++.

Responsibilities Overview: See next page for details

* + Andrew: Date\_Time, IO
  + Jake P.: List creation
  + Jake W.: List traversal functions (assume there exists a global HubNode \*head that points to the beginning of the hub linked list)
  + Jordan: HubNode, FlightNode (use UML diagram, without methods in FlightNode) (Due: Thursday)

Notes:

Jordan, please complete your assignment on Thursday, 3/20.

If you have any questions please contact Andrew or Jacob.

Class Overview:

* Date\_Time: stores time information for flights
  + Date\_Time(…): constructor w/ string parameters
  + AddMinutes(int min): used for calculating delays
  + ToString(): prints output, format on page 4 of specification
* HubNode: stores pointers to list of flights that leaves from hub and a pointer to the next hub
* FlightNode: stores flight information (number, price, etc.) and contains three pointer to source hub, destination of flight, and pointer to next flight in linked list for current hub
* Utilities
  + IO
  + List Creation
  + List Traversal

IO Functions:

* importHubs(pathToFile): read hub.csv, parse columns, and call addhubb(…) function. Structure while loop.
* importFlights(pathToFile): read flight.csv, parse columns, and call addFlight(…) function. Structure while loop.

List Creation

* addhub(string airport, string cityName)
* addFlight(string flightNumber, double price, string departure, int duration, string sourceAirportName, string destinationAirportName): Provide logic to determine flight company from flightNumber. Also call find hub for sourceAirportName and destinationAirportName. Also call Date\_Time using departure string. Create flight based on the flight company.

List Traversal Functions:

* findHub(string airportName): indexes through the linked list of hubs, searching for the hub based on the airportName string parameter. Returns pointer to hub.
* printFlights(): Traverses through every hub in the linked list, printing the hub information and traversing through every flight within that hub, printing the flight information
* deallocation(): traverses through every hub in the linked list, freeing every flight within that hub and then freeing the hub.