Computer Science and Engineering

Air Traffic Management System

System Requirements Specification (SRS)

Version 2.0

Document Number: SRS-002

Project Team Number: A11

Project Team Members:

Hriditaa Dekate (hrd259)

Faizan Hussain (fh828)

Neh Kundalia (Njk325)

Nick Tran (nt1456)

REVIEW AND APPROVALS

Team Members	Function (Author, Reviewer, Approval)	Date	Signature
Hriditaa Dekate	Author	10/7/20	Hriditaa Dekate
Neh Kundalia	Author	10/7/20	Neh Kundalia
Faizan Hussain	Author	10/7/20	Faizan Hussain
Nick Tran	Author	10/7/20	Nick Tran
Hriditaa Dekate	Author	10/19/20	Hriditaa Dekate
Neh Kundalia	Author	10/19/20	Neh Kundalia
Faizan Hussain	Author	10/19/20	Faizan Hussain
Nick Tran	Author	10/20/20	Nick Tran
Hriditaa Dekate	Reviewer	10/21/20	Hriditaa Dekate
Hriditaa Dekate	Author	11/19/20	Hriditaa Dekate
Neh Kundalia	Author	11/19/20	Neh Kundalia
Faizan Hussain	Author	11/19/20	Faizan Hussain
Nick Tran	Author	11/19/20	Nick Tran
Hriditaa Dekate	Reviewer	11/19/20	Hriditaa Dekate

REVISION LEVEL

Date	Revision Number	Purpose
10/8/20	Version 1.0	Initial Release
10/21/20	Version 2.0	Addition of Project Requirements
11/19/20	Version 3.0	Addition of System Model and Traceability, fixed previous defects

TABLE OF CONTENTS

1. DOCUMENT PURPOSE

1.1 Purpose

The purpose of the Software Requirements Specification document is to clearly lay out the system that will be developed, which is an Air Traffic Management System. The intended audience of this document is the airport management and air traffic controllers of John F. Kennedy Airport in New York, New York, who are the end users of the system. Other intended audience includes the project development team.

2. INTRODUCTION

2.1 Scope

The ATC is in need of an updated software system to reduce the delay from when an aircraft lands to when it taxis and reaches the gate.

T

he goal of ATMS will be to efficiently route the airplanes to the nearest available gates at the airport. The system will visualize the runways and the gates at various terminals to aid time efficiency. In case of an issue preventing a plane from reaching its designated gate, the system will efficiently reroute the plane to a different gate.

This system will not be responsible for relaying the relevant information to the passengers. It is intended to be used solely by the control system staff at the airport. The system is only meant to be used for planes that have either already landed or are about to take off. Thus, it is a ground airplane traffic control system.

2.2 Identification

Requirements Specification, SRS - 003, Revision 03

2.3 Bounds

The system shall work with the air traffic control system already in place and use the data received to make informed decisions to manage landings and ground navigation. It shall also provide the relevant flight information to the broadcast system which shall then relay that information to the passengers. The system shall be limited to use in the air traffic control office only.

2.4 Objectives

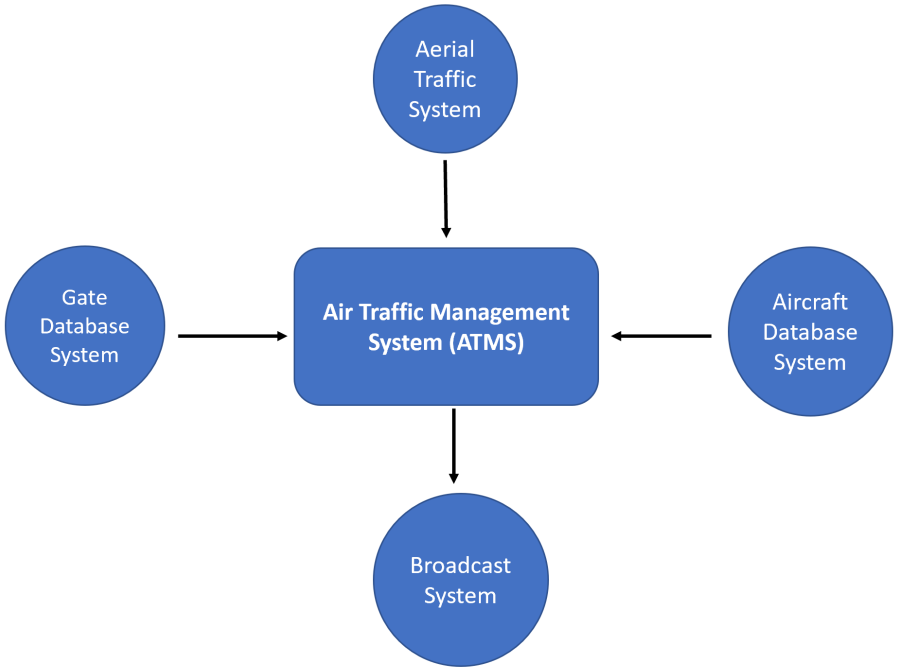
This project will be using the Agile Development Model. Efficiency in assigning gates and the safety of the people on board the aircraft shall be top priority. The UI of the system shall be medium priority as functionality is more important.

The project will produce incremental deliverables as the project progresses. The milestone dates are as follows:

Deliverable	Delivery Date
Project Proposal	September 22, 2020
Software Specifications Requirement(SRS) - Domain Definition	October 8, 2020
SRS - Project Requirements	October 22, 2020
Software Project Management Plan (SPMP)	November 5, 2020
SRS - Project Analysis	November 19, 2020

Presentation	December 15, 2020

2.5 Context Diagram



The system shall use data gathered from a database containing all relevant information about the gates, a database specifying which airplanes are currently present at the airport and the aerial traffic control system to make decisions about assigning planes to appropriate gates. The relevant information, after being processed, shall be sent to the broadcast system which will be in charge of relaying said information to the passengers.

2.6 Additional Descriptive Items

a. Product functions

The system automatically routes planes to respective gates based on availability and distance. It handles all the planes already on the ground and automatically guides planes to their correct

locations on the taxiways in order to remove manual intervention. It also reroutes airplanes to different destinations in case of unforeseen circumstances.

The system also periodically sends updated information to the airport broadcast system detailing changes made, if any.

b. User characteristics

This project is intended to be used by air traffic controllers at JFK airport. They will have administrative access to the software and hence are the controllers as well as prime users of the software. While the software automates the process of assigning planes to various gates, the user needs to be highly trained to understand and verify the processes done by the software. The user needs to know the layout of the airport, plane sizes and safety guidelines, air traffic protocols as well as airport procedures.

c. Constraints

One of the limitations of the system will be FAA regulations. The system needs to pass all the requirements as outlined by FAA. The system will also be limited to managing grounded aircraft. It will also depend on the date being provided by the aerial control system to be accurate. Although the system will aim to be as efficient as possible, the safety of passengers and pilots would be the top priority and efficiency may need to be sacrificed in lieu of the safety of the passengers.

d. Dependencies: The OS should be up to date as a security measure.

e. Requirements subsets – Frontend, Backend and Database. The frontend subset will be delayed as it has a lower priority.

3. GLOSSARY

ATC: Air Traffic Control

ATMS: Air Traffic Management System

OS: Operating System

SQA: Society of Quality Assurance

4. REFERENCE DOCUMENTS

A11 Project Proposal, Dekate, Hussain, Kundalia, Tran, v1.0, October 2020

A11 Software Management Plan, Dekate, Hussin, Kundalia, Tran, v1.0, November 2020

5. BUSINESS REQUIREMENTS

5.1 Technology

The main technology driver for this system is to provide air traffic controllers with a singular application that can be used to manage ground traffic at JFK airport. Planes will be automatically routed to gates and runways to maximize efficiency. Travelers will experience shorter wait times while taxiing for takeoff.

5.2 Economics

The demand for this system comes from the difficulty of scheduling airplane departure, arrival, delays, and cancellations. Having a central system that is accessible by airlines, airport managers, and travelers ultimately increases the efficiency of all parties involved by reducing fuel usage because it reduces wait times.

5.3 Regulatory and Legal

The system needs to pass all the requirements as outlined by FAA. Since this software works with aircraft carrying passengers safety shall be the number one priority. There is a high potential risk for legal action if the system is not capable of preventing outside attacks.

5.4 Market Considerations

Current air traffic management products do not automatically route airplanes to the nearest gates in order to reduce traffic and be able to accommodate unforeseen circumstances that can occur while taxiing from the runway to gates. Additionally, current products do not automatically communicate these changes to the pilots to be broadcasted to travelers.

5.5 Risks and Alternatives

No.	Title	Estimated Likelihood of Occurrence (0-10)	Estimated Impact	Estimated Cost of Management	Priority Number	Retirement Plan	Responsible Person	Target Completion Date
1	High Speed Impacts	<1	10	5	1	Suspend System and Evaluate	Hriditaa	Rolling Basis
2	Low Speed Impacts	<1	9	5	2	Suspend System and Evaluate	Neh	Rolling Basis
3	System Downtime	3	6	5	3	Cloud	Nick	Rolling Basis
4	Unauthorized Access	1	5	2	4	Monitor Access	Faizan	Rolling Basis

5.6 Human Resources and Training

Air traffic controllers need to be able to verify routes and changes created by the software. The developers need to have a firm understanding of the way JFK's ATC operates in order to create a useful software that will not hinder the performance of air traffic controllers.

6. USER REQUIREMENTS (DESCRIPTIVE FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS)

6.1 Functional Descriptive Detailed Requirements

1. The system shall allow users to monitor schedules that are created by the automated system.
    1. The user should be able to quickly report errors in the schedules.

2. The shall allow users to view the current location of all aircraft on the ground at all times.

a. The user should be able to search for an aircraft and see which gate the airplane is or will be located at.

3. The user should be able to manually change the gate, taxiway, or runway to use.

. The system shall require an identity verification before allowing a manual change.

a. The system shall ensure that the manual changes being made do not jeopardize the safety of any aircraft. It shall reject the changes if this happens to be the case.

4. The system shall be able to communicate directly with aircraft at all times.

. The user shall have the ability to communicate quickly with aircraft to determine the best possible schedule.

6.2 Non-Functional Descriptive Detailed Requirements

1. Product Requirement - Security
    1. The Air Traffic Management System shall only accept changes by authorized users within the Air Traffic Control Center.
    2. The system shall encrypt all communications with other systems to avoid a data leak.

2. Product Requirement - Performance (fast updates to schedules)
    1. The Air Traffic Management System shall provide updated ground traffic schedules quickly if a change is endured.
    2. The System shall always be up and running and should not have a downtime of more than 2 seconds in a day.
3. External Requirements - Regulatory and Safety
    1. The Air Traffic Management System shall function in accordance with Federal Aviation Administration (FAA) regulations.

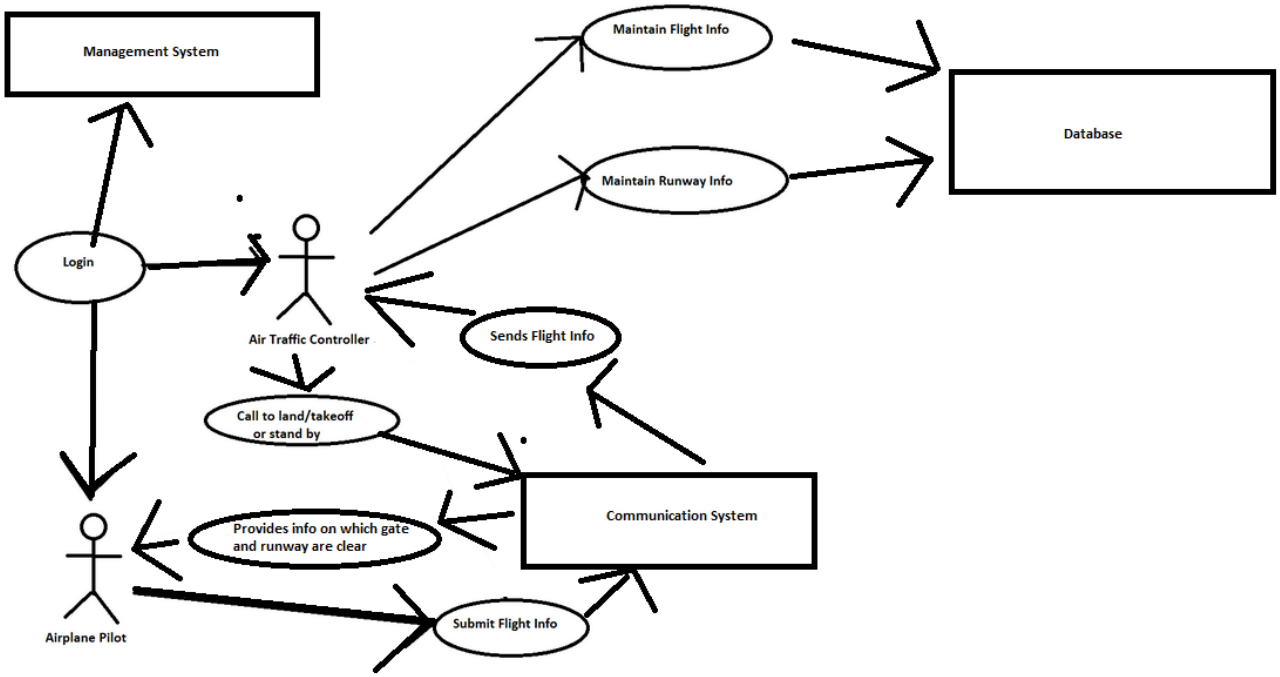
7. SYSTEM ARCHITECTURE

1. Management system - This part of the system shall act as a central interface for the user to interact with the software, view available data and to make changes to the schedule if necessary.
2. Database - The database will store all the necessary information (e.g., data about the location of all aircraft, available gates, outgoing flight data) needed in order to efficiently route airplanes.
3. Communication system - The communication system will provide users with an encrypted communication channel to send and receive messages to the aircraft as well as the airport broadcasting system.

8. DETAILED SYSTEM REQUIREMENTS – USE CASES

8.1 Requirement Use Cases

8.1.1 Use Case Diagrams



8.1.2 Use Case Descriptions

Login		
Descriptions	The ATC and airplane pilot will both log into the system in order to establish connection so that they can communicate.	
Pre-conditions	The ATC and airplane pilot must know the password and be in each other's vicinity.	
Flows	Basic or Normal Flows	1. ATC and airplane pilot both successfully login.

	Alternative Flows	1. If either party does not know the password they won't be able to access the system.
Post Conditions	The ATC will get in control of the system and send information and direct planes and the airplane pilot will be able to send their flight information to the ATC.	
Special Requirements	Knowledge of the password is required.	

Submit flight information		
Descriptions	The airplane pilot will submit its flight information: flight number, airline, departure destination, to the ATC.	
Pre-conditions	The ATC and airplane pilot must be logged in and in the vicinity of each other.	
Flows	Basic or Normal Flows	1. Airplane pilot submits flight information to the ATC. 2. ATC receives the information and stores it in the system.
	Alternative Flows	1. Error in receiving flight information or receiving incorrect information, in which case the airplane pilot will be prompted to send the correct information.

Post Conditions	The ATC will receive the information and provide appropriate guidelines to the airplane pilot.
Special Requirements	Stable communication is required between the airplane and ATC for all steps of this process.

Call to land/takeoff or standby		
Descriptions	The ATC will communicate with the Airplane Pilot whether the plane is ready for takeoff/landing or needs to be on standby and wait for further instructions. If in the air, waiting for landing, the plane might circle around.	
Pre-conditions	The ATC and airplane pilot must be logged in.	
Flows	Basic or Normal Flows	1. ATC reviews flight info submitted by airplane. 2. ATC configures what action must be taken: takeoff or landing. 3. In case of landing, send a free gate number for the plane to dock at. 4. For takeoff, direct the airplane to the appropriate runway. 5. ATC relays information to airplane pilots.
	Alternative Flows	1. If the runway isn't clear, relay to the airplane to standby for further communication. 2. In case of sudden gate or runway reallocations, relay that information to the plane.

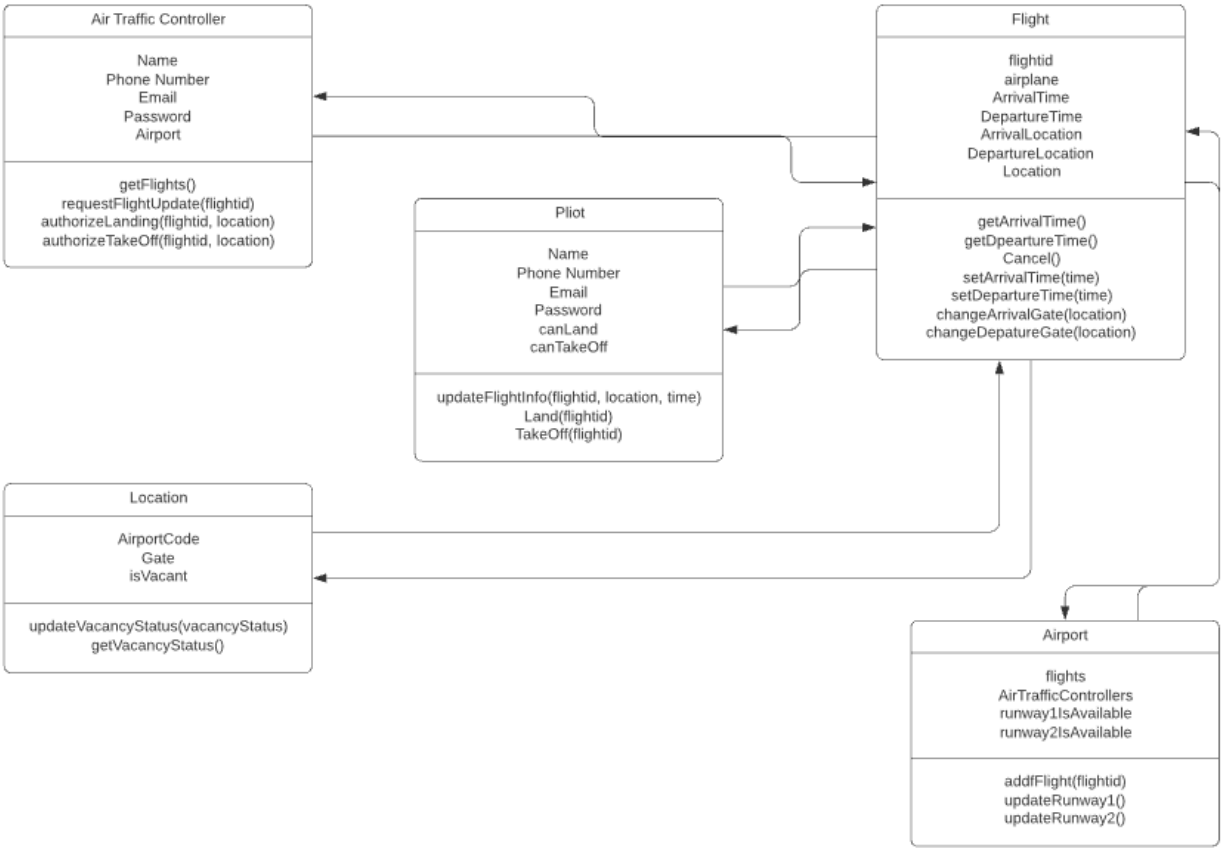
Post Conditions	The airplane pilot will see the information on what next action to take.
Special Requirements	Stable communication is required between the airplane and ATC for all steps of this process.

Maintain flight information		
Descriptions	The ATC will maintain information about all the planes at the airport as well as those in the vicinity in air.	
Pre-conditions	The ATC must be logged in and have appropriate information to store.	
Flows	Basic or Normal Flows	1. ATC maintains flight information of all airplanes. 2. ATC updates any information about any airplane as and when needed.
	Alternative Flows	1. Discard information when the plane loses vicinity of the airspace.
Post Conditions	The ATC database will be updated.	
Special Requirements	Stable access of the ATC to the database is required.	

Maintain runway information		
Descriptions	The ATC will maintain information about the runways as well as all the planes on the runways.	
Pre-conditions	The ATC must be logged in and have appropriate information about the airplanes.	
Flows	Basic or Normal Flows	1. ATC maintains runway information. 2. ATC updates any information about the different airplanes as they move on/off the runway.
	Alternative Flows	1. Discard information when the plane leaves the runway.
Post Conditions	The ATC database will be updated.	
Special Requirements	Stable access of the ATC to the database is required.	

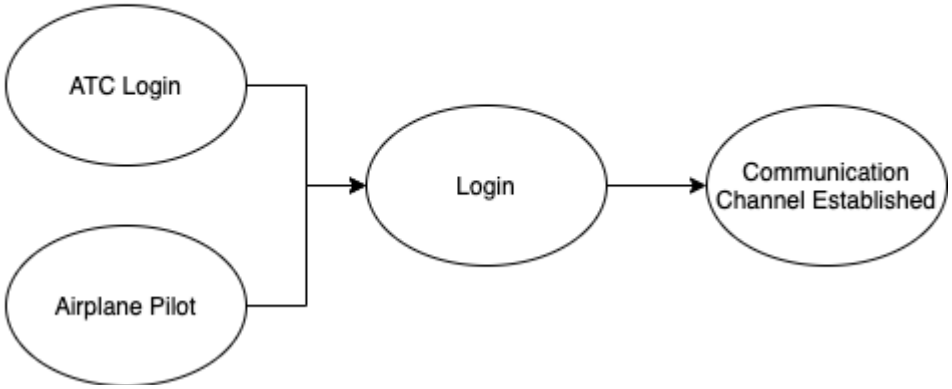
9. SYSTEM MODEL (UML)

9.1 Static - Class Diagrams

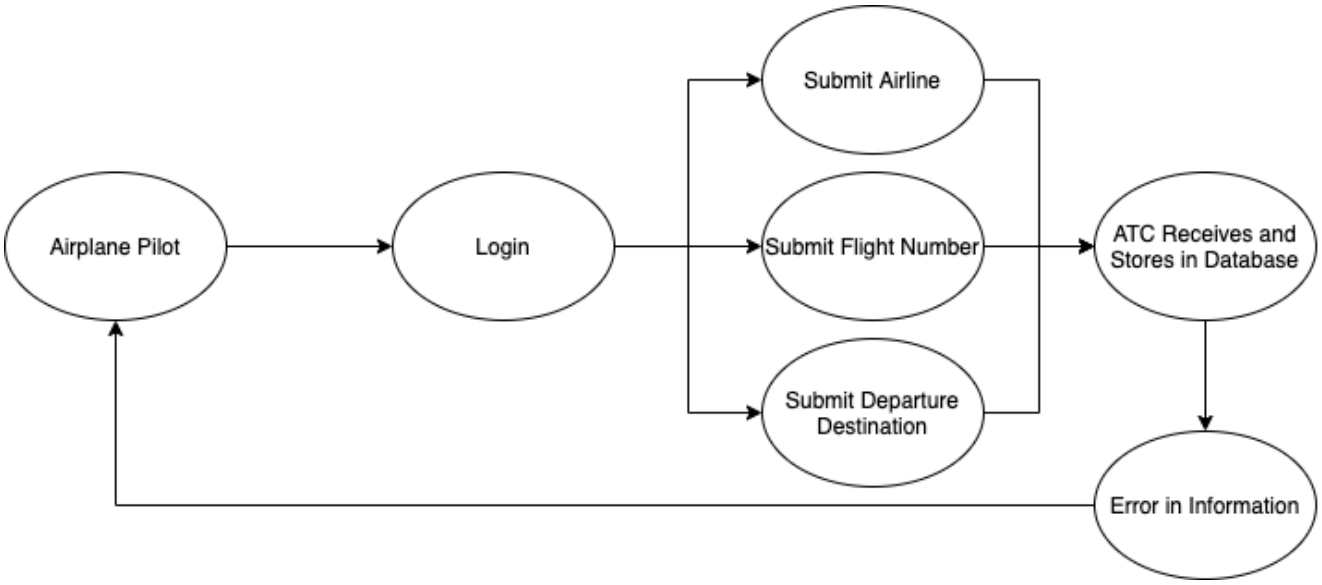


9.2 Dynamic - Behavioral Models

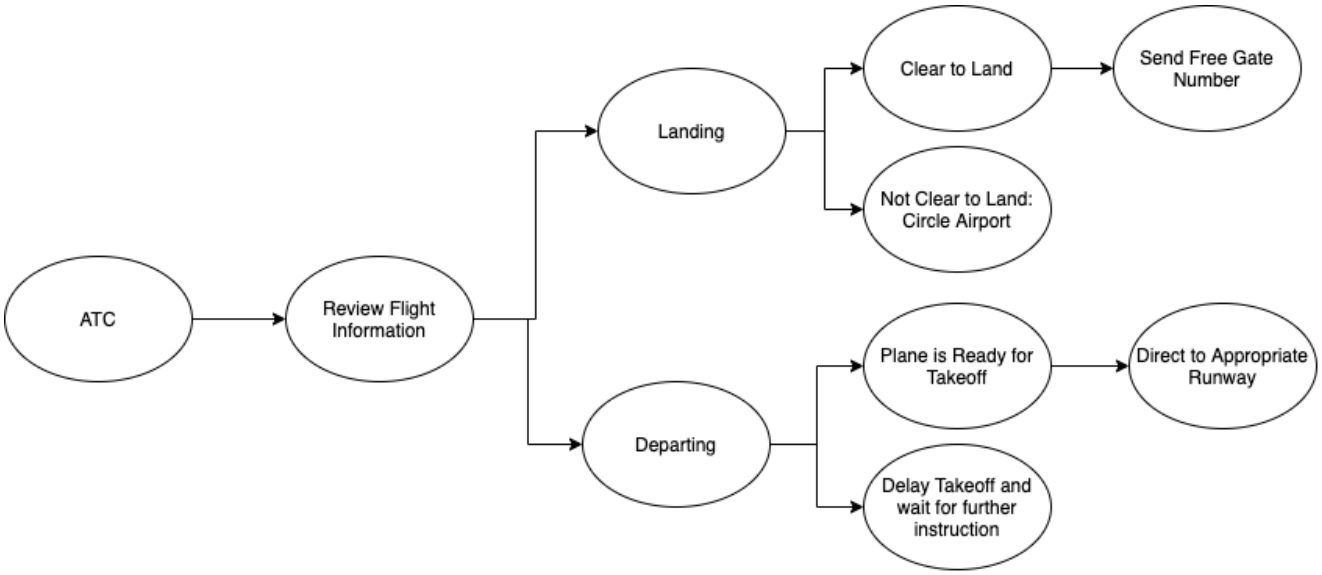
1. Login



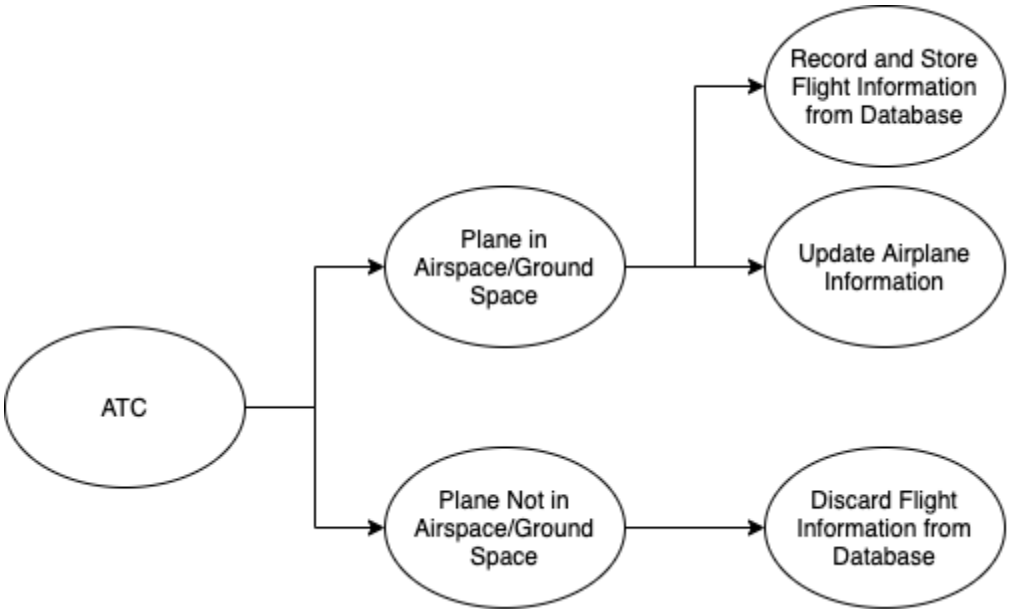
2. Submit Flight Information.



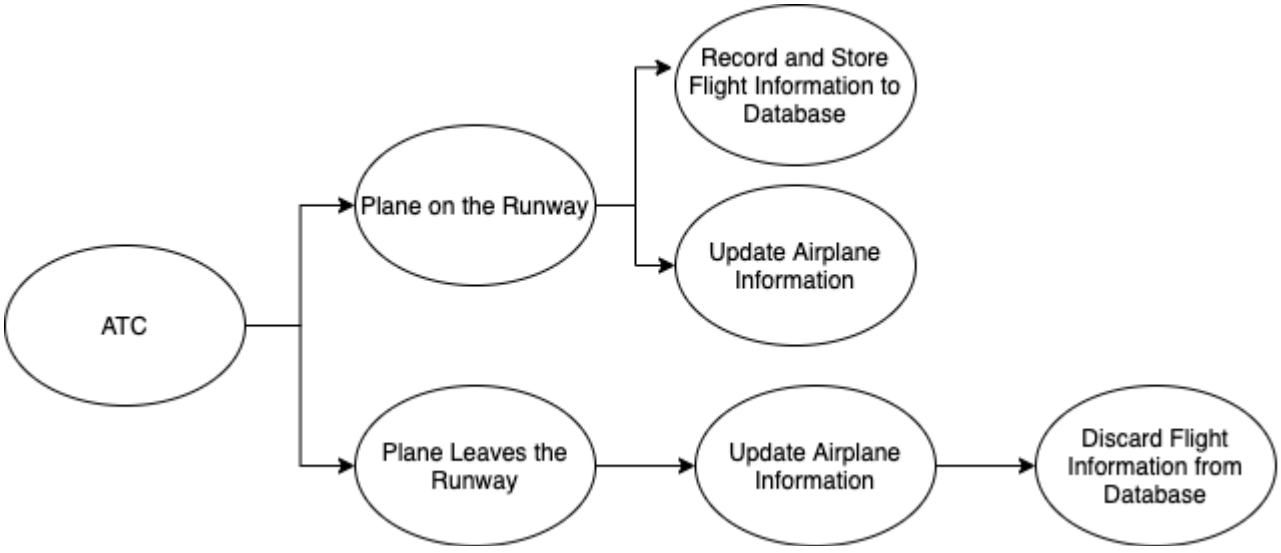
3. Call to Land/Takeoff or Standby



4. Maintain Flight Information



5. Maintain Runway Information



10. EVOLUTION OF THE SRS

We will update the document if the requirements are changed, when deliverables are completed, and if there are inaccuracies found within the document. If a requirements change needs to be made, a group member shall raise it with the group. From there a discussion within 7 days of the report shall arise on whether the change is needed. If 3 members of the group agree with a proposed change, the change shall be considered approved and carried out. If 2 or less people approve of the proposed change, then the change proposal shall be dismissed. In addition, every 2 weeks we shall convene and check if there are any proposed changes that any group member would like to make, and proceed with the previously mentioned procedure if so.

11. RATIONALE

None as of now

12. NOTES

None as of now

13. APPENDICES

13.1 System Test Plan Requirements

SQA testing process will be used to ensure the fulfillment of the requirements. The relevant use cases will be simulated for validation, which in this case involves testing with flight information of airplanes flying in and out of JFK, and attempting to see how the system manages gates. The validation process will measure performance based on a predetermined standard for the test.

The system shall be reliable. To that effect, the test will validate if the system ensures that planes do not collide, as any collisions would constitute a significant failure for an air traffic management system. The reliability of the system compared to human traffic controllers will also be tested to see how quickly the system can manage planes in and out of the airport, and the project shall ensure that the time it takes for an airplane to enter and leave an airport does not exceed 150% of the time it would take for the plane under human traffic control. The management system will be tested for varying levels of air traffic.

Testing shall maintain the integrity of the system by validating its cybersecurity through hacking attempts to ensure that traffic management remains secure and subject only to the purview of legitimate air traffic controllers.

13.2 Qualification Provisions

The document will be reviewed through individual and peer review for individual changes to ensure that the document maintains correctness, unambiguity, completeness, stability, and verifiability. Group review will take place with regards to defects and on a biweekly basis as specified in Section 10. If a proposal is made under a group review and accepted, the next group review will

address the progress of the proposal, and allocate resources as necessary to aid or terminate the proposal.

13.3 Requirements Traceability

All requirements that are stated in this document will be traceable. Traceability will be executed on the numbered order of each requirement as defined in Section 6. The requirements stated in section 6 can be referred to as Function Requirement (Number) or Nonfunctional Requirement (Number), to be traced both forwards and backwards.

13.4 Schedule Tracking

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 1.0 – Project Domain Definitions	Faizan Hussain	2 hours	2 hours	0 hours
SRS Version 1.0 – Project Domain Definitions	Hriditaa Dekate	2 hours	2 hours	0 hours
SRS Version 1.0 – Project Domain Definitions	Neh Kundalia	2 hours	2 hours	0 hours
SRS Version 1.0 – Project Domain Definitions	Nick Tran	2 hours	2 hours	0 hours
SRS Version 1.0 – Project Domain Definitions	Total	8 hours	8 hours	0 hours

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 2.0 – Requirements	Faizan Hussain	2 hours	0.5 hours	-1.5 hours

SRS Version 2.0 – Requirements	Hriditaa Dekate	2 hours	1.5 hours	-0.5 hours
SRS Version 2.0 – Requirements	Neh Kundalia	2 hours	2 hours	0 hours
SRS Version 2.0 – Requirements	Nick Tran	2 hours	2 hours	0 hours
SRS Version 2.0 – Requirements	Total	8 hours	6 hours	-2 hours

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SPMP Version 1.0	Faizan Hussain	4 hours	4 hours	0 hours
SPMP Version 1.0	Hriditaa Dekate	3 hours	3.5 hours	0.5 hours
SPMP Version 1.0	Neh Kundalia	4 hours	4 hours	0 hours
SPMP Version 1.0	Nick Tran	4 hours	3.5 hours	-0.5 hours
SPMP Version 1.0	Total	15 hours	15 hours	0 hours

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 3.0 - Analysis	Faizan Hussain	3 hours	4 hours	1 hours
SRS Version 3.0 - Analysis	Hriditaa Dekate	5 hours	4 hours	-1 hours
SRS Version 3.0 - Analysis	Neh Kundalia	4 hours	4 hours	0 hours
SRS Version 3.0 - Analysis	Nick Tran	4 hours	3.5 hours	-0.5 hours
SRS Version 3.0 - Analysis	Total	16 hours	15.5 hours	-0.5 hours

Cumulative

Who (individual or Team)	Estimated	Actual	Difference
Faizan Hussain	11 hours	10.5 hours	-0.5 hours
Hriditaa Dekate	12 hours	11 hours	-1 hours
Neh Kundalia	12 hours	12 hours	0 hours
Nick Tran	12 hours	11 hours	-1 hours
Total	47 hours	44.5 hours	-2.5 hours

13.5 Defect Tracking

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 1.0 – Project Domain Definitions	Faizan Hussain	3	4	1
SRS Version 1.0 – Project Domain Definitions	Hriditaa Dekate	1	2	1
SRS Version 1.0 – Project Domain Definitions	Neh Kundalia	4	4	0
SRS Version 1.0 – Project Domain Definitions	Nick Tran	2	1	-1
SRS Version 1.0 – Project Domain Definitions	Total	10	11	1

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 2.0 – Requirements	Faizan Hussain	1	1	0
SRS Version 2.0 – Requirements	Hriditaa Dekate	2	2	0
SRS Version 2.0 – Requirements	Neh Kundalia	5	4	-1
SRS Version 2.0 – Requirements	Nick Tran	3	5	2
SRS Version 2.0 – Requirements	Total	11	12	1

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SPMP Version 1	Faizan Hussain	4	3	-1
SPMP Version 1	Hriditaa Dekate	3	5	2
SPMP Version 1	Neh Kundalia	1	0	-1

SPMP Version 1	Nick Tran	6	4	-2
SPMP Version 1	Total	14	12	-2

Artifact or Deliverable	Who (individual and team)	Estimated	Actual	Difference
SRS Version 3 - Analysis	Faizan Hussain	4	4	0
SRS Version 3 - Analysis	Hriditaa Dekate	5	5	0
SRS Version 3 - Analysis	Neh Kundalia	3	3	0
SRS Version 3 - Analysis	Nick Tran	4	4	0
SRS Version 3 - Analysis	Total	16	16	0

Cumulative

Who (individual and team)	Estimated	Actual	Difference
Faizan Hussain	12	12	0
Hriditaa Dekate	11	14	3
Neh Kundalia	13	11	-2
Nick Tran	15	14	-1
Total	41	41	0

13.6 Dictionaries

Class

Name	Description	Methods	Attributes
Air Traffic Controller	This class represents the person who will be handling the flight schedule management and assignment	getflights() requestFlightUpdate(flightid) authorizeLanding(flightid,location) authorizeTakeOff(flightid,location)	Name PhoneNumber Email Password Airport
Pilot	This class represents the pilots who will be communicating with the Air Traffic	updateFlightInfo(flightid,location,time) land(flightId) takeOff(flightid)	Name PhoneNumber Email

	Controller		Password CanLand CanTakeOff
Flight	This class represents the flights that might be landing, taxiing or taking off.	getArrivalTime() getDepartureTime() Cancel() setArrivalTime(time) changeArrivalGate(location) changeDepartureGate(location)	flightid airplane ArrivalTime DepartureTime ArrivalLocation DepartureLocation Location
Location	This represents the individual gates at the airport	updateVacancyStatus(vacancyStatus) getVacancyStatus()	AirportCode Gate IsVacant
Airport	This represents the airport at which the planes will be landing and taking off	addFlight(flightid) updateRunway1() updateRunway2()	flights AirTrafficControllers runway1isAvailable runway2isAvailable

Methods

Name	Description	Class	Arguments
getFlights()	For the air traffic controller to be able to see all the flights	Air Traffic Controller	NA

requestFlightUpdate()	To see the current flight status of a particular flight	Air Traffic Controller	flightid
authorizeLanding()	To authorize landing on the runway for a particular flight	Air Traffic Controller	flightid, location
authorizeTakeoff()	To authorize takeoff on the runway for a particular flight	Air Traffic Controller	flightid, location
updateFlightInfo()	For the pilot to be able to make any changes mid-flight if necessary	Pilot	flightid, location, time
Land()	for the pilot to indicate that the plane has landed	Pilot	flightid
TakeOff()	for the pilot to indicate that the plane is taken off	Pilot	flightid
getArrivalTime()	to be able to see arrival time of the flight	Flight	NA
getDepartureTime()	to be able to see arrival time of the flight	Flight	NA
Cancel()	to indicate if a flight is cancelled	Flight	NA
setArrivalTime()	to change arrival time to a different time	Flight	time
setDepartureTime()	to change departure time to a different time	Flight	time
changeArrivalGate()	to change gate of arrival	Flight	location
change Departure Gate()	to change gate of departure	Flight	location

updateVacancyStatus()	to change the status gate availability	Location	vacancyStatus
getVacancy()	to get current status of the gate	Location	NA
addFlight()	to add new flight record at the airport	Airport	flightid
updateRunway1()	to change the availability status of runway 1 (empty or full)	Airport	NA
updateRunway2()	to change the availability status of runway 2 (empty or full)	Airport	NA

Attributes

Name	Description	C/S	Type	Size	R/W
GateList	**An array of Gates**	**Simple**	**Array**	**200**	**R/W**
FlightList	**An array of Flights**	**Simple**	**Array**	**1000**	**R/W**
PilotList	**A list of pilots**	**Simple**	**Array**	**1000**	**R/W**
Name	**Represents a name**	**Simple**	**String**	**100**	**R/W**
PhoneNumber	**Represents phone number**	**Simple**	**String**	**10**	**R/W**
Email	**Represents email**	**Simple**	**String**	**50**	**R/W**
Password	**Represents encrypted password**	**Simple**	**String**	**25**	**R/W**

CanLand	**Represents whether a plane can land**	**Simple**	**Boolean**	**1**	**R/W**
CanTakeOff	**Represents whether a plane can take off**	**Simple**	**Boolean**	**1**	**R/W**
flightid	**Represents the unique id of a flight**	**Simple**	**Integer**	**1**	**R**
airplaneid	**Represents the unique id of an airplane**	**Simple**	**Integer**	**1**	**R**
ArrivalTime	**Represents the arrival time of a flight at the airport**	**Simple**	**Time**	**1**	**R**
DepartureTime	**Represents the departure time of a flight from the airport**	**Simple**	**Time**	**1**	**R/W**
AirportCode	**A code that uniquely identifies the airport**	**Simple**	**String**	**10**	**R**

Relationship

Name	Description	From class	To class	Optional/ Mandatory	Cardinality
Communicate	The AirTraffic Controller will communicate with incoming flights.	AirTraffic Controller	Flight	M	one-to-many
IsPilotOf	Each flight will have a relationship to its pilot.	Pilot	Flight	M	one-to-one
IsAt	If a flight lands at a gate, this relationship will be established to keep track of flights	Flight	Location	O	one-to-one
IsAt	The flight may land at the airport at which point this relationship is established.	Flight	Airport	O	many-to-one

Key Events

Name	Description	Motive	Action	Pre-conditions	Post conditions	State Change
Login	Pilot and ATC log into the system to establish connection	Necessary to maintain safety and security of the system	Entering password into the system	The ATC and airplane pilot must know the password and be in each other's vicinity.	The ATC will get in control of the system and send information and direct planes and the airplane pilot will be able to send their flight information to the ATC.	Yes
Submitting flight information	The airplane pilot will submit its flight information: flight number, airline, departure destination, to the ATC.	Necessary so that ATC can keep track of current information of all flights	Sending relevant information	The ATC and airplane pilot must be logged in and in the vicinity of each other.	The ATC will receive the information and provide appropriate guidelines to the airplane pilot.	Yes
Call to land/takeoff	The ATC will communicate with the Airplane Pilot whether the plane is ready for takeoff/landing or needs to be	Necessary to guide the flight to make the right move on the airport	Sending relevant information to the pilot	The ATC and airplane pilot must be logged in.	The airplane pilot will see the information on what next action to take.	Yes

	on standby and wait for further instructions. If in the air, waiting for landing, the plane might circle around.					
Providing runway and gate information	The ATC will communicate with the Airplane Pilot which runway and gate should the plane proceed to.	Necessary to guide safe navigation of plane at the airport	Send relevant information	The ATC and airplane pilot must be logged in.	The airplane pilot will see the information on what next action to take.	Yes

14. INDEX

User Requirements, Pg. 9