Title:	**Create Account**
Description:	Allow a pilot user to create an account that registers them as a pilot.
System Under Design:	The system here is the Pilot Account Management system, which handles all functionality and data display that a pilot user requires
Primary Actor:	Pilot user
Participants:	FireStore Database
Goal:	To have a pilot account that they can then authenticate with to access their data
Following Use Cases:	Log In
Invariant:	NA
Precondition:	User must not already be authenticated
Success Postcondition:	A persistent account is created in the database which the pilot can log into with the credentials they created

STEPS:	**ALTERNATIVES**:
1. Put credentials in signup page 2. Click Sign up button	

Title:	**Add Plane Specs**
Description:	A page that allows a pilot to add the specs of their plane to their account so that passengers viewing it may see what kind of plane they have
System Under Design:	The system here is the Pilot Account Management system, which handles all functionality and data display that a pilot user requires
Primary Actor:	Pilot user
Participants:	FireStore Database
Goal:	Allow the addition of place specifications to the database
Following Use Cases:	View Specs, Change Specs
Invariant:	User must be authenticated as a pilot
Precondition:	User must be authenticated with a pilot account
Success Postcondition:	Spec Data is stored persistently in the database

STEPS:	**ALTERNATIVES**:
1. Click “add specs” button 2. Fill out form with specs 3. click apply button	

Title:	**Select Airports**
Description:	Allows a pilot to select what airports they want to show up at when passenger users go looking for rides
System Under Design:	The system here is the Pilot Account Management system, which handles all functionality and data display that a pilot user requires
Primary Actor:	Pilot User
Participants:	FireStore Database
Goal:	To allow a pilot to mark airports which they are willing to get to in order to pick up passengers
Following Use Cases:	Find Pilot, book trip
Invariant:	User must be authenticated as a pilot
Precondition:	User must be authenticated as a pilot
Success Postcondition:	Database must persistently store what pilots are at what airports

STEPS: 1. Click “add airport availability button” 2. search for airports you want to add to 3. select the airport 4. click save changes button	**ALTERNATIVES**:

Title:	**Make Payment**
Description:	Allow a passenger user to make a payment to a pilot user
System Under Design:	Account Interaction System which handles all functionality that involves sharing data between user accounts
Primary Actor:	passenger user
Participants:	pilot user
Goal:	Allow passengers to pay pilots
Following Use Cases:	accept payment

Invariant:	The trip being paid for must not be canceled, and both user accounts must not be deleted or become disallowed from financial transactions.
Precondition:	The passenger must have booked a trip with the pilot in question
Success Postcondition:	The pilot must receive the money from the passenger

STEPS:	**ALTERNATIVES**:
1. Click “make payment” button 2. Display total cost 3. Click “confirm payment” button 4. Route money to pilot	

Title:	**Contact Passengers**
Description:	A text conversation is started between a pilot user and a passenger user, using our service as a middle man so that both parties may communicate without having to share their phone numbers.

System Under Design:	Account Interaction System which handles all functionality that involves sharing data between user accounts
Primary Actor:	Pilot user
Participants:	Passenger User
Goal:	Allow communication between pilots and passengers who are in the same trip route without giving away personal information.
Following Use Cases:	None
Invariant:	Neither account contact information must change and the trip must not be canceled.
Precondition:	Both accounts must be authenticated
Success Postcondition:	A communication chanel is set up between pilot and passenger without exchanging personal information

STEPS:	**ALTERNATIVES**:
1. Click “start conversation” button. 2. Conversate.	

Title:	**Create Account**
Description:	Allow a passenger user to create an account that registers them as a passenger
System Under Design:	The system this use case is under is the Passenger Account Management system, which handles the functionality and data that a passenger user requires
Primary Actor:	Passenger user
Participants:	FirseStore Database
Goal:	to have a passenger account s that they can authenticate it and access their data related to their account
Following Use Cases:	Log In
Invariant:	NA
Precondition:	User must not already be authenticated
Success Postcondition:	An account is created in the database which the passenger can log into with the credentials they created

STEPS: 1. Enter credentials into the signup page 2. Click the sign up button	**ALTERNATIVES**:

Title:	**Contact Pilot**
Description:	A text conversation is started between the pilot user and passenger user, using our service as a middle man so that both parties are communicating without providing their phone number if they don’t want to.
System Under Design:	The Account Interaction System which handles functionality that involves sharing data between user accounts.
Primary Actor:	Passenger users.
Participants:	Pilot user.
Goal:	Allow communication between pilots and passengers without providing personal information (phone number).
Following Use Cases:	none
Invariant:	Neither account contact information can change and the trip must not be canceled
Precondition:	both accounts must be authenticated
Success Postcondition:	a communication vessel is set up between pilot and passenger without exchanging personal information

STEPS: 1. click “start conversation” button 2. Converse	**ALTERNATIVES**:

Title:	**Locate Pilots**
Description:	Available pilots are located based on the users recorded location and preferred airports
System Under Design:	The system this use case is under is the Passenger Account Management system, which handles the functionality and data that a passenger user requires
Primary Actor:	Passenger user
Participants:	none
Goal:	Allow users to have a variety of pilots to choose from in their area, giving the passenger user a choice of pilots near them to take a trip with
Following Use Cases:	Book flight Contact Pilot

Invariant:	NA
Precondition:	Must have input zip code and selected preferred airports
Success Postcondition:	Multiple pilots in the area are revealed to the user

STEPS:	**ALTERNATIVES**:
1. Click locate pilots 2. View nearby pilots	

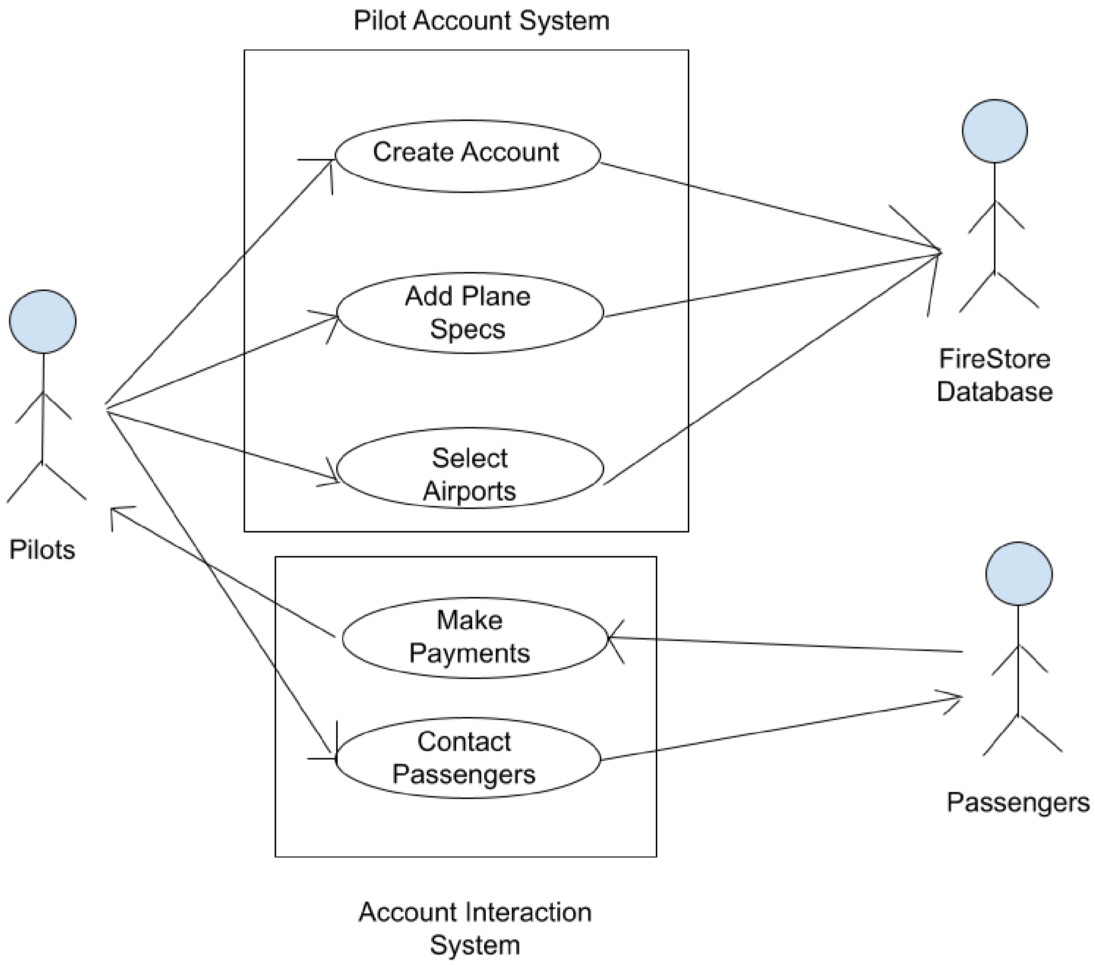
Title:	**Book Flights**
Description:	A future flight is scheduled after both users agree upon a fare, allowing the passenger to get ready for their trip and allowing the pilot to make the appropriate preparations for the aircraft

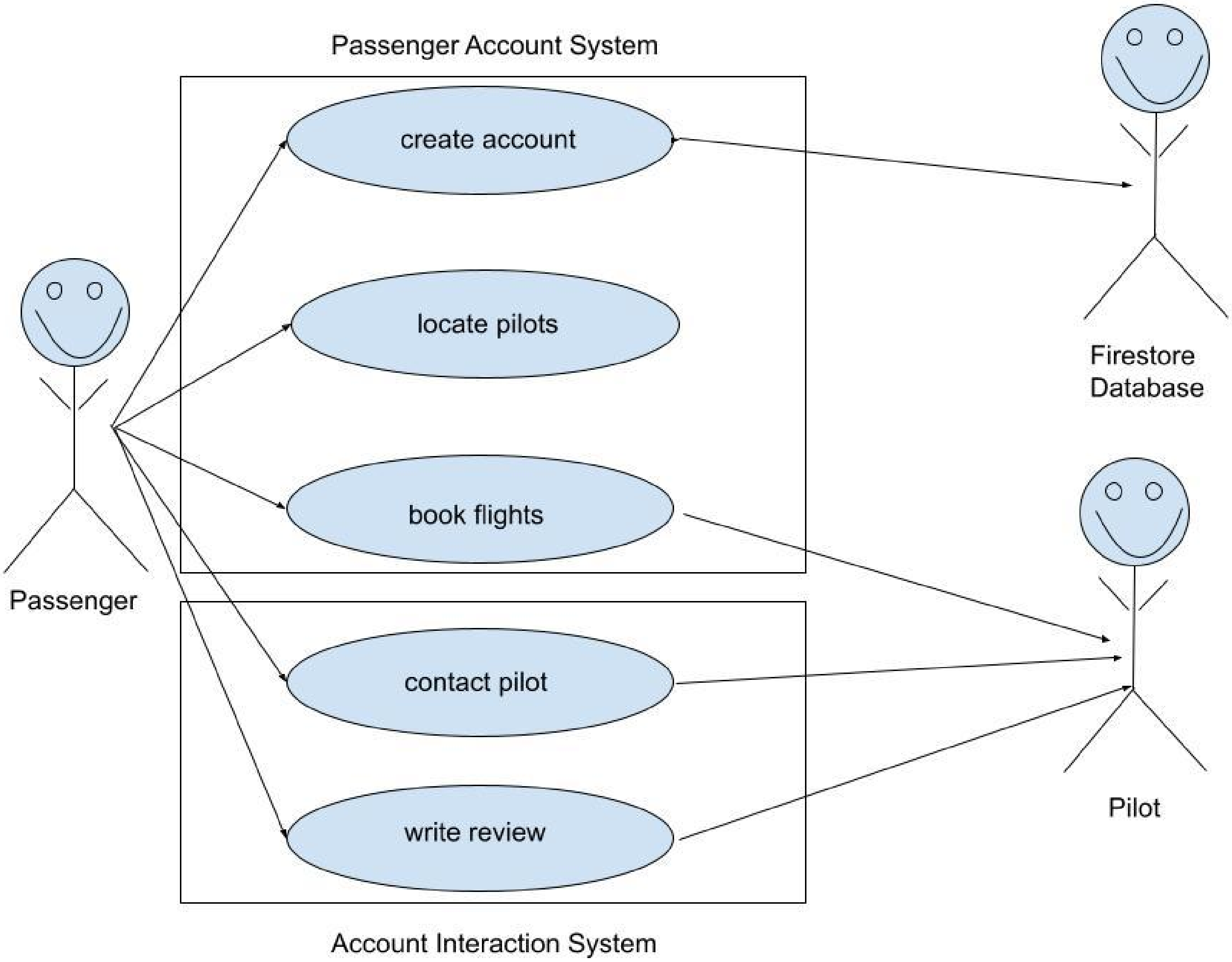
System Under Design:	The system this use case is under is the Passenger Account Management system, which handles the functionality and data that a passenger user requires
Primary Actor:	Passenger User
Participants:	Pilot User
Goal:	Users are able to schedule a flight for a trip they would like to take in the near future
Following Use Cases:	Make payment
Invariant:	Neither account contact information must change and users must be communicating
Precondition:	Both accounts must be authenticated
Success Postcondition:	A flight is booked and both users are aware of the flight date

STEPS:	**ALTERNATIVES**:
1. Communicate with pilot 2. Agree upon flight date 3. Click the “Book Flight” button	

Title:	**Write Review**
Description:	A review is sent to the Pilot director about the passengers experience on the trip, and said review is added to the database
System Under Design:	The Account Interaction System which handles functionality that involves sharing data between user accounts.
Primary Actor:	Passenger user
Participants:	Pilot User Firestore Database
Goal:	Allow for other passenger users to see reviews of pilots before booking a flight with them, also allows pilot user to take feedback from the passenger user so that they can provide the best experience possible
Following Use Cases:	none
Invariant:	N/A
Precondition:	Both accounts must be authenticated, and a booked flight must have been completed
Success Postcondition:	Passenger users are able to tell the pilot and other passengers who will in the future be using this pilot about their experience.

STEPS: 1. Click “write a review” 2. Provide a reiveiw	**ALTERNATIVES**:


Pilot Account System
Create Account
Add Plane Specs
Select Airports
FireStore Database
Pilots
Make Payments
Contact Passengers
Passengers
Account Interaction System


Passenger Account System
create account
locate pilots
book flights
contact pilot
write review
Account Interaction System
Passenger
Firestore Database
Pilot