**Requirement Analysis**

**Angela Mattamana, Michael Richter, Charles Nguyen, Paul Blackburn**

**Use Case 1: Register trip**

**Primary Actor:** Driver

**Offstage Actor:** Rider

**Stakeholders and Interests:**

* Driver: want an easy way to make an advertisement for their trip
* Rider: want the necessary information from the driver to decide if they want to join them

**Preconditions:** Driver is identified and registered an account with the application.

**Success Guarantee:** Trip is registered. Email confirmation of the trip. Other riders can now view the trip and decide if they want to join. The driver will receive updates regarding his trip via email.

**Main Success Scenario:**

1. Driver opens the application and clicks on the “create trip” button.

2. Driver enters the appropriate trip details such as start time, start location, final destination, number of seats available, etc into text fields.

3. Driver inputs payment information for the rider to send payment towards.

3. Driver reviews the information and clicks on the “submit” button to finalize the trip.

4. The rider now has access to the created trip on their feed and can decide to join.

5. Driver waits for the riders to respond to the newly created trip.

**Extensions:**

2a. Trip Details

1. Driver can customize their profile picture to display alongside the trip details.

2. Driver can choose their own preferred music from a selection of defined generes.

3. Driver fails to enter a required field when creating a trip.

a. Driver enters the field properly and is able to complete creating the trip.

**Special Requirements:**

System updates with new trip details within 30 seconds 90% of the time

**Frequency of Occurrence:**

This occurs only once per trip at the start to initialize the process.

**Use Case 2: Driver Score**

**Primary Actor:** Rider

**Offstage Actor:** Driver

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Driver and rider are registered users of the app and have been validated as Virginia Tech students.

**Success Guarantee:** Rating for Driver is accepted after drop off. Overall rating for Driver is updated and shown on profile.

**Main Success Scenario:**

1. Driver drops off rider at requested drop off location.

2. Driver indicates trip completed with driver on app

3. System confirms by checking location of driver.

4. Driver Feedback page pops up on app of rider.

5. Rider gives Driver a rating out of 5 stars.

6. Rider provides any additional feedback in a text entry box if any.

7. Rider submits the Driver Feedback form.

8. System uses the submitted rating to update overall rating for driver and reflects it on the driver’s profile page.

9. System submits the additional comments to the app team and they take any necessary action depending on the comments.

**Extensions:**

9a. Negative Comments

1. System signals recent review and notifies the driver that the review is negative.

2. Driver responds to the negative comment:

a. Driver reports the comment as false

i. Driver presses the report comment button

ii. System identifies that the user has reported the comment

iii. System confirms the review with the rider

3. Driver receives an investigation after repeated negative reviews.

**Special Requirements:**

Driver score updates within 30 seconds 90% of the time

App team takes a close look at feedback provided

**Technology and Data Variation List:**

3a. Location accuracy using driver’s phone service

**Frequency of Occurrence:**

Every time a driver drops off a rider

**Use Case 3: Choosing riders**

**Primary Actor:** Driver

**Offstage Actor:** Rider

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Driver and rider are registered users of the app and have been validated as Virginia Tech students. Additionally, the driver has submitted a trip and has received feedback from various interested riders.

**Success Guarantee:** The driver can choose between the riders by swiping to accept and swiping to decline based upon the prompted rider profile.

**Main Success Scenario:**

1. Driver receives a notification that people are interested in his trip.

2. Driver opens the application and goes to the matching section for his posted trip.

3. Driver now is prompted with the information of an interested rider.

4. Driver decides to accept the rider and adds them to their trip.

5. The rider receives a notification that they have been approved for the trip they were interested in.

6. The Driver and the rider are put in a group chat to facilitate communication between the members of the trip.

7. System updates the members of the trip and removes the trip from the market if all the seats are full

**Extensions:**

4a. Choosing a rider

1. Driver denies the rider based upon their profile removing them from the applicant pool.

2. Driver wants to know more information about the rider before making a decision.

a. Driver sends a message clarifying information stated by the rider.

i. Rider confirms the information and the driver accepts the rider.

ii. Rider does not respond to the driver and is not added to the trip.

6a. Driver and Rider group chat

1. Driver sends a message stating that they have limited trunk space.

2. Driver sends a message stating that there is a chang in pickup time and location.

a. Rider sends a message regarding their response to this change of information

i. Rider removes themselves from the trip due to conflict.

ii. Rider agrees to the changes of the trip and stays on board.

**Special Requirements:**

System updates the members of the trip within 30 seconds 90% of the time

**Technology and Data Variation List:**

2a. Algorithm to prompt the best fit between the rider and driver first when the driver is selecting between riders.

**Frequency of Occurrence:** Every time a new rider is interested in the trip the driver posted.

**Use Case 4: Driver is speeding**

**Primary Actor:** Rider

**Offstage Actor:** Driver

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Driver and rider are undergoing a trip to an agreed upon destination.

**Success Guarantee:** Rating for Driver is negatively impacted by conducting traffic infractions. The driver receives fines internally from the application sent to the rider that dealt with the poor driving.

**Main Success Scenario:**

1. Driver increases their speed to a reckless level due to not paying attention to the speed limit.

2. Rider opens the application and clicks “Report Driver Speed”

3. System confirms by checking location of driver and speed of the driver.

4. Driver Speed Feedback page pops up on app of rider confirming the report of speed.

5. Driver completes the trip and drops off the rider.

6. The driver receives a notification afterwards regarding penalties identified by the rider.

7. Driver pays for the penalties and sends the money towards the rider.

8. System confirms the payment and notifies the rider of the successful completion of the report.

**Extensions:**

3a. System confirmation Fails

1. System signals that the driver was not speeding and will not receive penalties for their actions.

2. System blocks the rider for making too many reports within a set window of time.

7a. Driver doesn’t pay their fines.

1. Driver receives a fixed amount of interest per fine that they don’t pay by the required date. Ultimately, the driver would be sued if they didn’t adhere to the payments.

**Special Requirements:**

System notifies the rider of successful completion of the report within 30 seconds 90% of the time

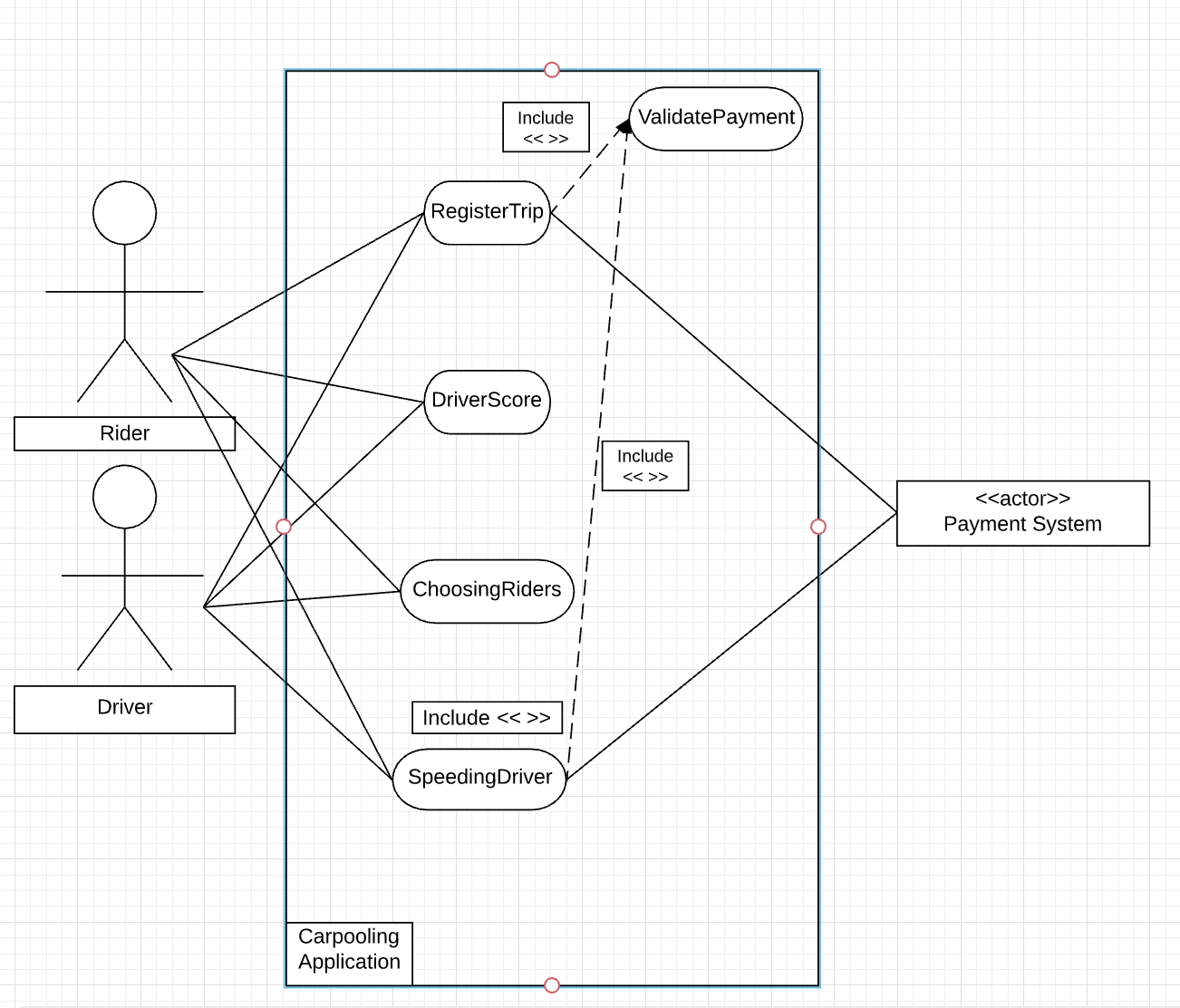
**Technology and Data Variation List:**

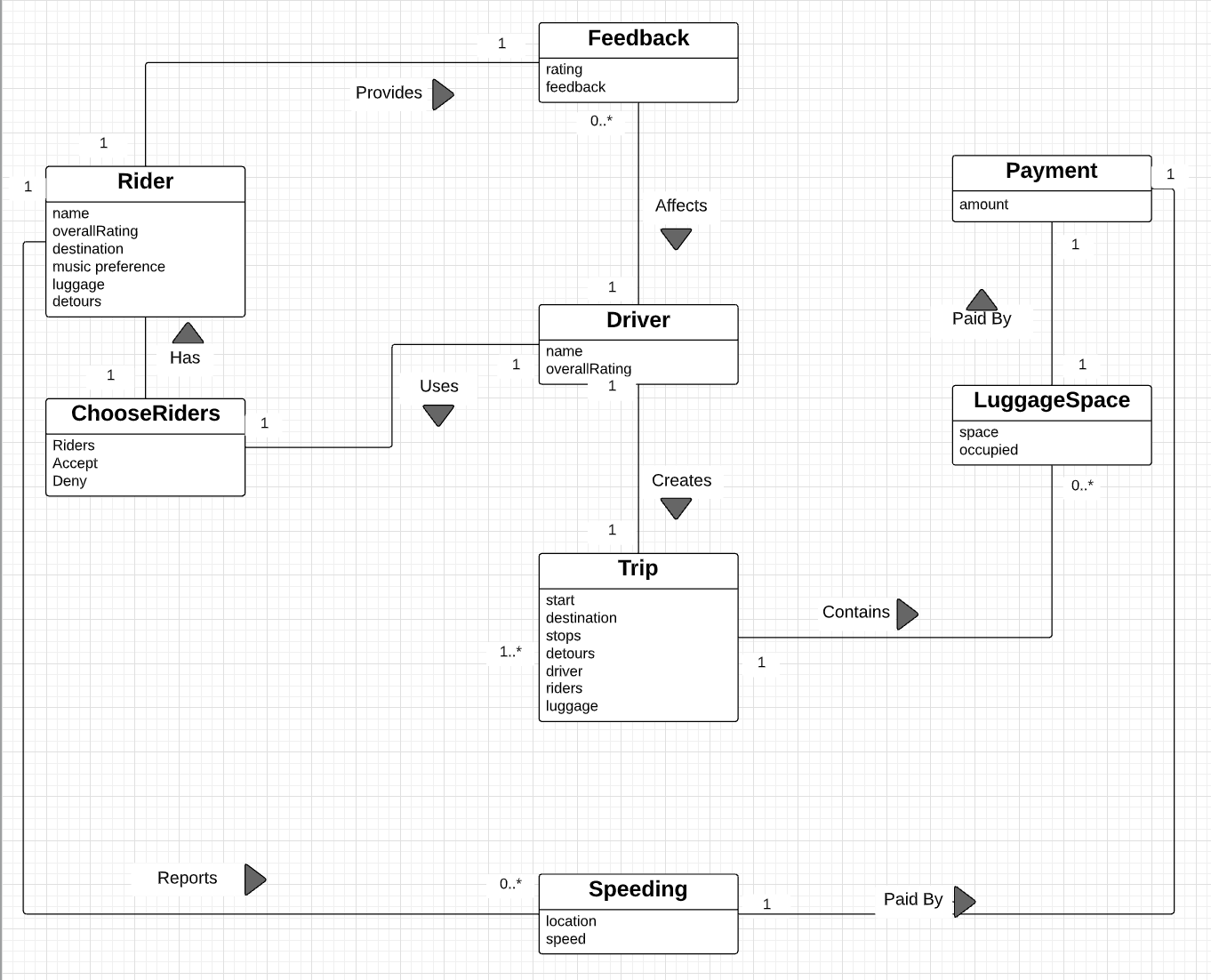
7a. Technology to send payments between driver and rider.

**Frequency of Occurrence:**

When a driving infraction is reported by a rider.

**Use Case Diagram and Conceptual Class Diagram for Use Cases 1 - 4**





**Use Case 5: Rider Score**

**Primary Actor:** Driver

**Offstage Actor:** Rider

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Driver and rider are registered users of app and have been validated as Virginia Tech students.

**Success Guarantee:** Rating for rider is accepted after drop off. Overall rating for rider is updated and shown on profile.

**Main Success Scenario:**

1. Driver drops off rider at requested drop off location.

2. Driver indicates trip completed with driver on app

3. System confirms by checking location of driver.

4. Rider Feedback page pops up on app.

5. Driver gives rider a rating out of 5 stars.

6. Driver provides any additional feedback in a text entry box if any.

7. Driver submits the Rider Feedback form.

8. System uses the submitted rating to update overall rating for rider and reflects it on the rider’s profile page.

9. System submits the additional comments to the app team and they take any necessary action depending on the comments.

**Extensions:**

3a. Location not Verified

1. System signals error and notifies driver that current location was not the agreed upon location with rider

2. Driver responds to the error:

a. Driver’s current location was not updated

i. Driver presses button indicating to refresh driver’s current location

ii. System updates driver’s current location

iii. System confirms current location of driver is the agreed upon drop off location

b. Drop off location was verbally changed with driver

i. Driver presses button indicating drop off location was changed to current location.

ii. System sends requests to rider asking if the current location is the new drop off location

iii. Rider indicates yes.

**Special Requirements:**

Rider score updates within 30 seconds 90% of the time

App team takes a close look at feedback provided

**Technology and Data Variation List:**

3a. Location accuracy using driver’s phone service

**Frequency of Occurrence:**

Every time a driver drops off a rider

**Use Case 6: Rider must be able to purchase luggage space**

**Primary Actor:** Rider

**Supporting Actor:** Payment Company (such as Venmo, Hokie Passport, etc.)

**Offstage Actor:** Driver

**Stakeholders and Interests:** Rider, Driver, Payment Company

**Preconditions:** Driver has accepted rider for ride and vice versa.

**Success Guarantee:** Request for additional luggage allocation was accepted and total space for luggage in car was decreased.

**Main Success Scenario:**

1. Rider opens app and clicks on the existing ride.

2. Rider clicks on Purchase Additional Space For luggage to purchase additional space for luggage other than the already allocated space (luggage that can fit on rider’s lap or by his/her feet)

3. Options for additional luggage space show up on screen.

4. Rider chooses one or more different options that best fits their needs.

5. Rider confirms choice and amount to be paid.

6. System takes rider to payment page.

7. Rider chooses payment method.

8. System validates payment method with payment company and system handles payment.

9. System logs completed sale and presents electronic receipt.

10. Purchased additional luggage space is shown on existing ride.

**Extensions:**

7a. Rider wants to use another payment method.

1. Rider clicks Add Payment Method.

2. Rider enters required information.

3. System validates information and adds payment method to existing payment methods.

4. System handles payment.

8a. System cannot communicate with payment company

1. Failure log is recorded.

2. System communicates failure to communicate with payment company to rider.

3. Rider is prompted to try again later.

8b. Payment method does not have sufficient funds to pay for additional luggage

1. Failure log is recorded.

2. System communicates no sufficient funds to pay for additional luggage in selected payment method to rider.

3. Rider is prompted to select from options to handle this.

a. Select another payment method

i. System takes rider back to screen to choose payment methods.

ii. Rider chooses another payment method.

iii. System handles payment.

b. Add another payment method

i. Rider enters required information.

ii. System validates information and adds payment method to existing payment methods.

iii. System handles payment.

c. Try again later after adding sufficient funds to payment method.

i. Rider is prompted to try again later

**Special Requirements:**

System updates luggage space within 30 seconds 90% of the time

System logs failures

**Frequency of Occurrence:**

Whenever a rider feels they need additional luggage space other than their seat

**Use Case 7: Search for rides based on proximity to desired final destination**

**Primary Actor:** Rider

**Offstage Actor:** Driver

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Rider is registered user of app and has been validated as Virginia Tech student.

**Success Guarantee:** Rider finds options for rides to desired final destination.

**Main Success Scenario:**

1. Rider opens app.

2. Rider presses Find Rides button.

3. Rider enters desired final destination.

4. System uses algorithm to search for best options for drivers offering rides based on final destination.

5. One option shows up on screen.

6. Rider is allowed to look at driver profile and ride details.

7. Rider swipes left if they are not interested in the ride and right if they are interested

*Repeat 5-7 until options are over or rider indicates done.*

8. Options rider selected shows up on rider’s Current Rides tabs until further communication from drivers.

**Extensions:**

4a. System cannot find any rides based on entered final destination

1. System signals error and notifies rider that no rides were found.

2. Rider responds to error:

a. Rider is prompted to enter another final destination if they want.

b. System logs desired final destination

i. When any driver makes a posting in the future about a ride that is close to rider’s desired final destination, system sends notification to rider.

ii. Rider can contact driver about ride.

c. Rider is prompted to try again later.

**Special Requirements:**

A list of options based on proximity is received within 30 seconds of submission of desired final destination 90% of the time

System logs failures

**Technology and Data Variation List:**

4.a Location accuracy services to best figure out which destinations are closer than others

**Frequency of Occurrence:**

When riders need to find a trip to or from Virginia Tech

**Use Case 8: Submit requests for detours**

**Primary Actor:** Rider

**Offstage Actor:** Driver, Other riders in the car

**Stakeholders and Interests:** Rider, Driver, Other riders in the car

**Preconditions:** Driver and rider agreed upon the ride.

**Success Guarantee:** Request submitted and the driver and other riders in the car get notified or request.

**Main Success Scenario:**

1. Rider opens app and clicks on the current ride.

2. Rider clicks on Request for Detour.

3. Rider enters a location.

4. Rider enters a reason for detour request.

5. Rider presses submit.

6. System logs request for detour.

7. System sends request to the driver and other riders in car if there are any.

**Extensions:**

6.a System fails.

1. System logs failure and communicates this to rider.

2a. System prompts rider to try again later.

2b. System prompts rider to verbally communicate detour request to driver and other

riders in the car.

**Special Requirements:**

System logs failure

Request for detours are sent to driver and other riders within 30 seconds 90% of the time

**Frequency of Occurrence:**

When a rider feels like they need to make a stop during their trip for whatever reason

**Use Case 9: Rider can cancel ride up to 72 hours before departure**

**Primary Actor:** Rider

**Supporting Actor:** Payment companies, Hokie Passport, Venmo, Zelle

**Offstage Actor:** Driver

**Stakeholders and Interests:** Driver, Rider, Payment companies

**Preconditions:** Rider must have a payment account with sufficient funds to pay driver. Rider must cancel 72 hours before the departure time

**Success Guarantee:** The rider must cancel 72 hours in advance to ensure that the money is returned to the rider.

**Main Success Scenario:**

1. The rider requests a ride two weeks in advance to go home for Thanksgiving break
2. Money is put on hold when the application confirms that there are sufficient funds in the payment accounts
3. The driver accepts the request
4. Five days before the departure date, the riders’ parents tell them that they plan on coming to pick them up.
5. The rider confirms that the parents are coming to pick them up
6. The rider cancels the appointment 80 hours before the departure date
7. The rider’s money is returned to the payment account

**Extensions:**

3a. The driver already has all their seats filled up, so the rider can not get a seat in the car for the ride back

6a. The rider does not cancel the ride within time, they cancel it 60 hours before, but the application will not allow that

**Special Requirements:**

Access to check the balances of the payment accounts

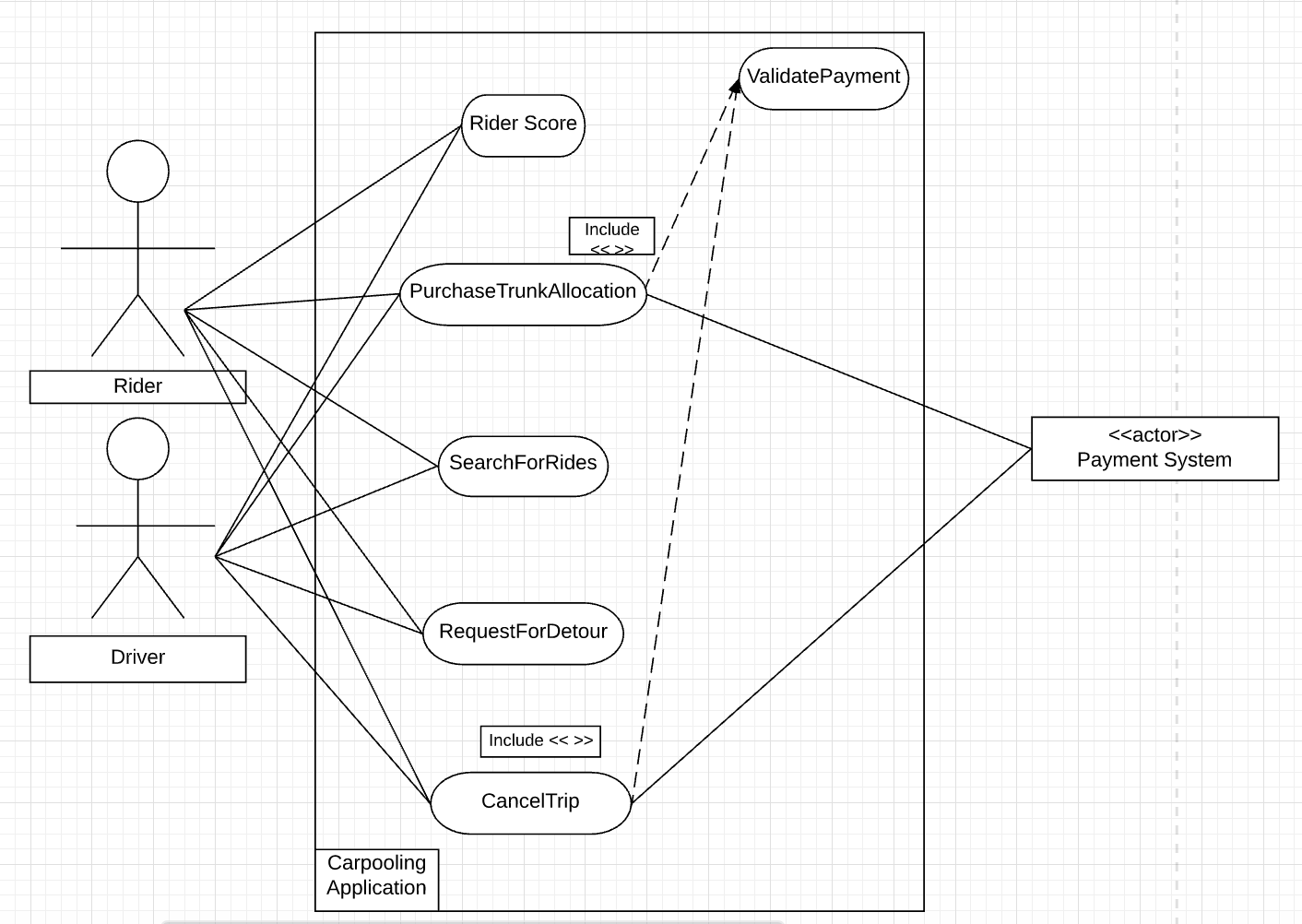
**Technology and Data Variation List:**

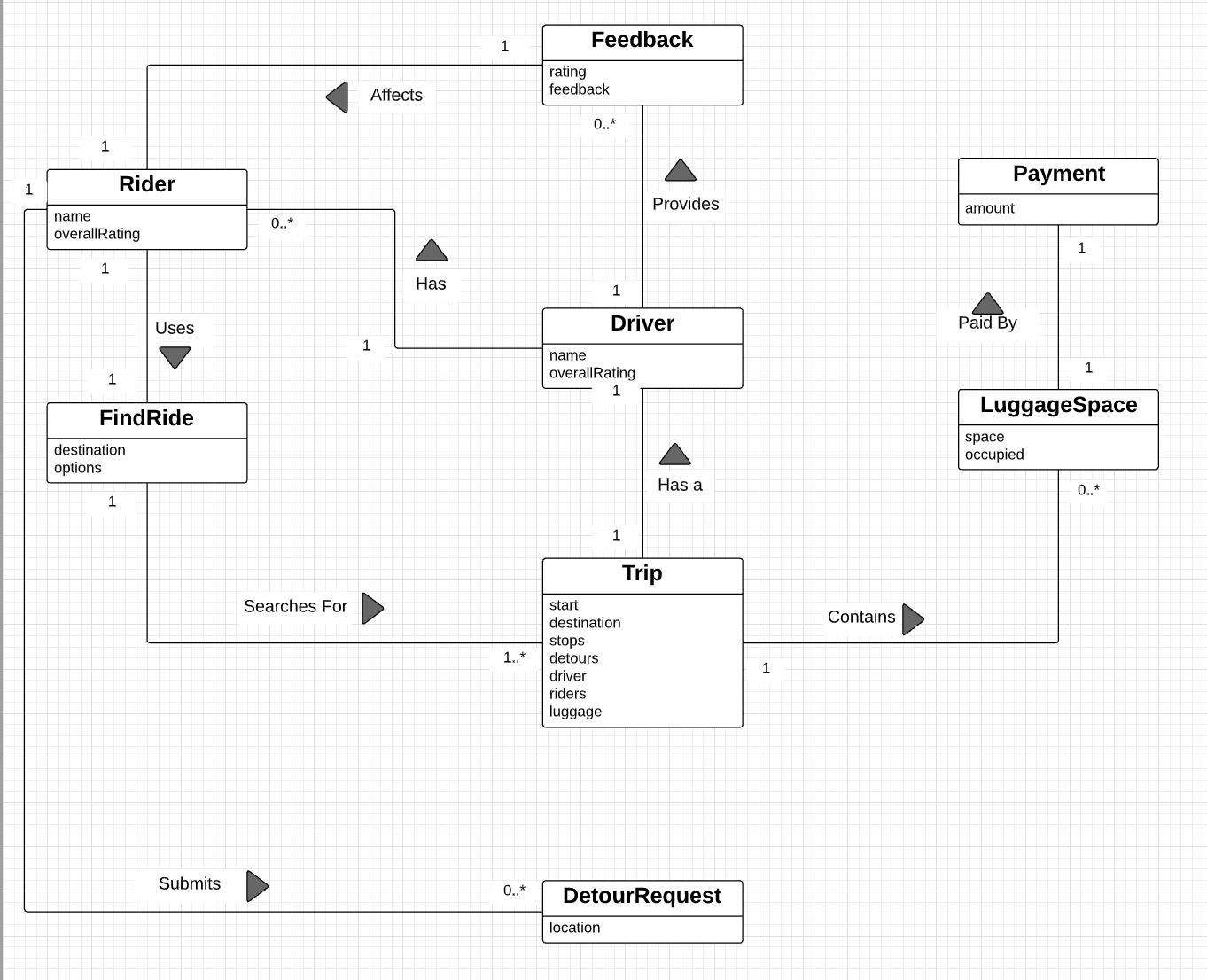
7a. Technology to return money to account

**Frequency of Occurrence:**

Everytime a rider requests a ride

**Use Case Diagram and Conceptual Class Diagram for Use Cases 5 - 9**

****

****

**Use Case 10: Riders sign up for app with Hokie Passport**

**Primary Actor:** Virginia Tech Student

**Offstage Actor:** Non-Virginia Tech Student

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Virginia Tech Student has their hokie passport

**Success Guarantee:** Virginia Tech Student signs up for the app

**Main Success Scenario:**

1. User opens the app.

2. User presses create account.

3. Application asks for permission to use the camera.

4. User snaps a photo of their Hokie Passport

5. Application parses the static text (name, birthday, and id) on the Hokie Passport.

6. Application pulls additional information (email) from VT Servers (<http://search.vt.edu/search/m/people.html>)

7. User enters any additional information.

8. Application submits user information to backend and the account is persistent for future sessions

**Extensions:**

4a. User snaps a picture of their drivers license (don’t have a Hokie Passport)

Prompt user to recapture

4a. User snaps a picture that is not in focus

Prompt user to recapture

**Frequency of Occurrence:**

Once per application download / (mobile app and or web app instance per user)

**Use Case 11: Drivers must sign up with valid license and insurance**

**Primary Actor:** Rider

**Offstage Actor:**

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Rider has a valid account (enabled the application to use the camera

**Success Guarantee:** Virginia Tech Student adds driver permissions to their account (previously they just had rider permission)

**Main Success Scenario:**

1. User opens the app (their session is persistent so they are automatically logged in).

2. User presses apply to drive.

3. User snaps a photo of their drivers’ license and insurance card

4. Application parses the static text (Driver’s License Number, Sex, and Insurance Policy Number) from the images.

5. Application submits user information to backend

6. Backend uses algorithm to give driver a safety score

Lower scores for males, points on license etc.

7. Backend automatically rejects applications with low safety score

8. System admin manually inspects applications before approving the driver

**Extensions:**

3a. User snaps a picture a picture of a piece of paper for their insurance card

Application fails to validate the insurance card and prompts user to recapture

3a. User snaps a picture that is not in focus

Recapture picture

**Technology and Data Variation List:**

4a. Card reader

6a. Safety score algorithm using data from their driver’s licence

**Frequency of Occurrence:**

Once per application download / (mobile app and or web app instance per user)

**Use Case 12: Remove accounts of riders and drivers who have graduated or engage in unsafe behavior**

**Primary Actor:** Rider, Driver

**Offstage Actor:**

**Stakeholders and Interests:** Driver, Rider

**Preconditions:**

**Success Guarantee:** The account is deactivated once the student graduates. The account is deactivated if the driver fails to drive the route

**Main Success Scenario:**

1. A backend nightly job queries the relational database for reported incidents.

2. Negative incidents are joined with the user table.

3. These user accounts are flagged as suspended

4. A suspended user attempts to use the app and receives a pop up their account has been suspended

**Extensions:**

1. A backend nightly job queries the relational database for users.

2. Check if user is no longer enrolled

3. These user accounts are flagged as suspended

4. A suspended user attempts to use the app and receives a pop up their account has been suspended

**Technology and Data Variation List:**

1a. Access to VT servers

**Frequency of Occurrence:**

Spring and Fall graduations

**Use Case 13: Present terms of service in an ethical manner**

**Primary Actor:** Rider, Driver

**Offstage Actor:** Lawyer, Technical writer

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Act Ethically

**Success Guarantee:** Drivers and Riders understand the terms of service because they are presented in a human readable format (concise) instead of legal jargon (small font, lengthy).

**Main Success Scenario:**

1. User signs up for rider account or adds driver permissions to their account

2. App presents terms of service as a slide deck of key points (large font)

3. User swipes left to read the next card (5-10 cards)

4. User can sign up for their account

**Extensions:**

1. User opens the app and clicks on terms of service

2. App presents terms of service as a slide deck of key points (large font)

3. User swipes left to read the next card (5-10 cards)

1. Receive terms of service document from a lawyer

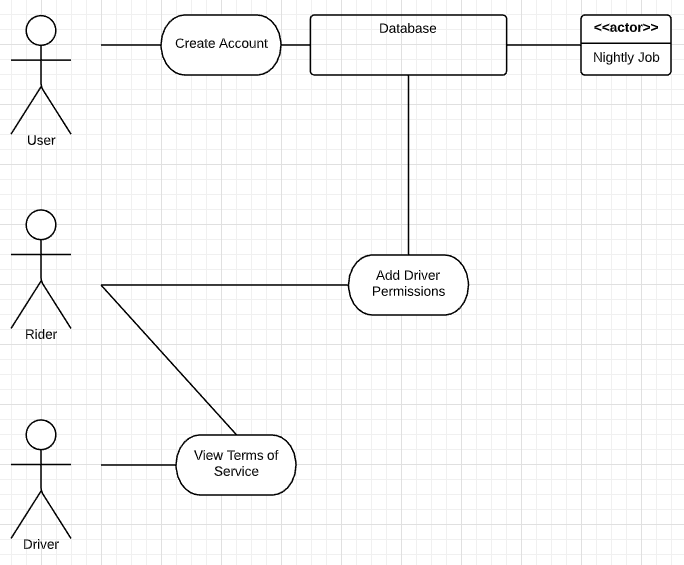
2. Technical writers translate legal jargon into layman's terms

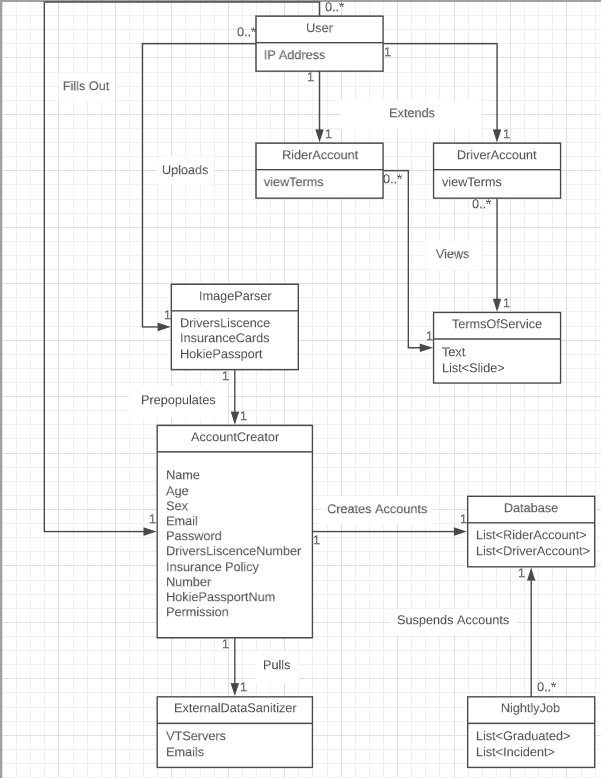
3. Front end designers make easy to read slide deck

**Frequency of Occurrence:**

Varies

**Use Case Diagram and Conceptual Class Diagram for Use Cases 10 - 13**





**Use Case 14: Support Hokie Passport, Venmo, and Zelle as valid payment methods**

**Primary Actor:** Rider

**Supporting Actor:** Payment companies, Hokie Passport, Venmo, Zelle

**Offstage Actor:** Driver

**Stakeholders and Interests:** Driver, Rider

**Preconditions:** Must have money in accounts to pay driver

**Success Guarantee:** Driver and riders understand the amount of money that it will cost for the ride, and the rider has sufficient funds to pay the driver.

**Main Success Scenario:**

1. Rider must be a Virginia Tech student to sign up for the application and be a valid rider.
2. Rider should have one of the following payment methods, Hokie Passport, Venmo, or Zelle
3. Rider will open the application and click on add payment
4. Rider will link their payment method with the application so that the driver can be paid
5. Rider will click submit payment method so that their account is updated with the payment method
6. Rider can now use the application because the payment method is linked to the account

**Extensions:**

1a. Rider is not a Virginia Tech student

2a. Rider needs to sign up for one of the payment methods or they must add more money to their Hokie Passport

5a. The information must be correct so that the payment method links properly to the application

**Frequency of Occurrence:**

Occurrences only when a new payment method needs to be updated after the initial addition of payment method.

**Use Case 15: Validate payments**

**Primary Actor:** Rider

**Supporting Actor:** Payment companies, Hokie Passport, Venmo, Zelle

**Stakeholders and Interests:** Driver, Rider, Payment companies

**Preconditions:** Must have a payment account with sufficient funds to pay driver

**Success Guarantee:** The rider has sufficient funds to cover the whole trip

**Main Success Scenario:**

1. The rider requests a ride.
2. The application checks to see that the rider has a valid payment account that can be withdrawn from.
3. The rider checks to see that they have enough money to cover the whole trip.
4. The application puts the money on hold when the rider requests a ride.
5. If there’s enough money in the payment accounts, the money can be put on hold to ensure there’s enough money when the ride is completed
6. The ride request goes through
7. The driver accepts the ride request
8. The rider gets a ride

**Extensions:**

1a. Rider is not a Virginia Tech student

2a. The rider could potentially not have an account linked up to pay the driver, so the rider must set up the payment method.

3a. If the rider does not have enough money in the balance, then the money can not be put on hold

5a. If the balance is too low in the proposed payment method, the user must add more money to the account so the ride process can continue,

**Special Requirements:**

Access to check the balances of the payment accounts

**Technology and Data Variation List:**

5a. Technology to check the balances

**Frequency of Occurrence:**

Everytime a rider requests a ride

**Use Case 16: Driver receives payment upon completion of ride**

**Primary Actor:** Rider, Driver

**Supporting Actor:** Payment companies, Hokie Passport, Venmo, Zelle

**Stakeholders and Interests:** Driver, Rider, Payment companies

**Preconditions:** Rider must have a payment account with sufficient funds to pay driver

**Success Guarantee:** The rider has sufficient funds to cover the whole trip and both the rider and driver accept the drive

**Main Success Scenario:**

1. The rider requests a ride.
2. The application checks the funds to make sure the rider has enough money to pay the driver
3. The driver accepts the ride
4. Any additional stops are requested before the ride begins
5. The driver drives the rider to the destination on the day and time that is set out for the appointment
6. The rider confirms that they are at their location
7. The driver confirms that they are at their location
8. The money that has been put on hold when the application checked to see if the rider had enough money goes to the drivers account
9. If there are multiple riders, each of the riders have individual payments that they have to be charged, and the driver gets money separately from each of the riders

**Extensions:**

1a. Rider is not a Virginia Tech student

2a. The rider could potentially not have an account linked up to pay the driver, so the rider must set up the payment method.

**Special Requirements:**

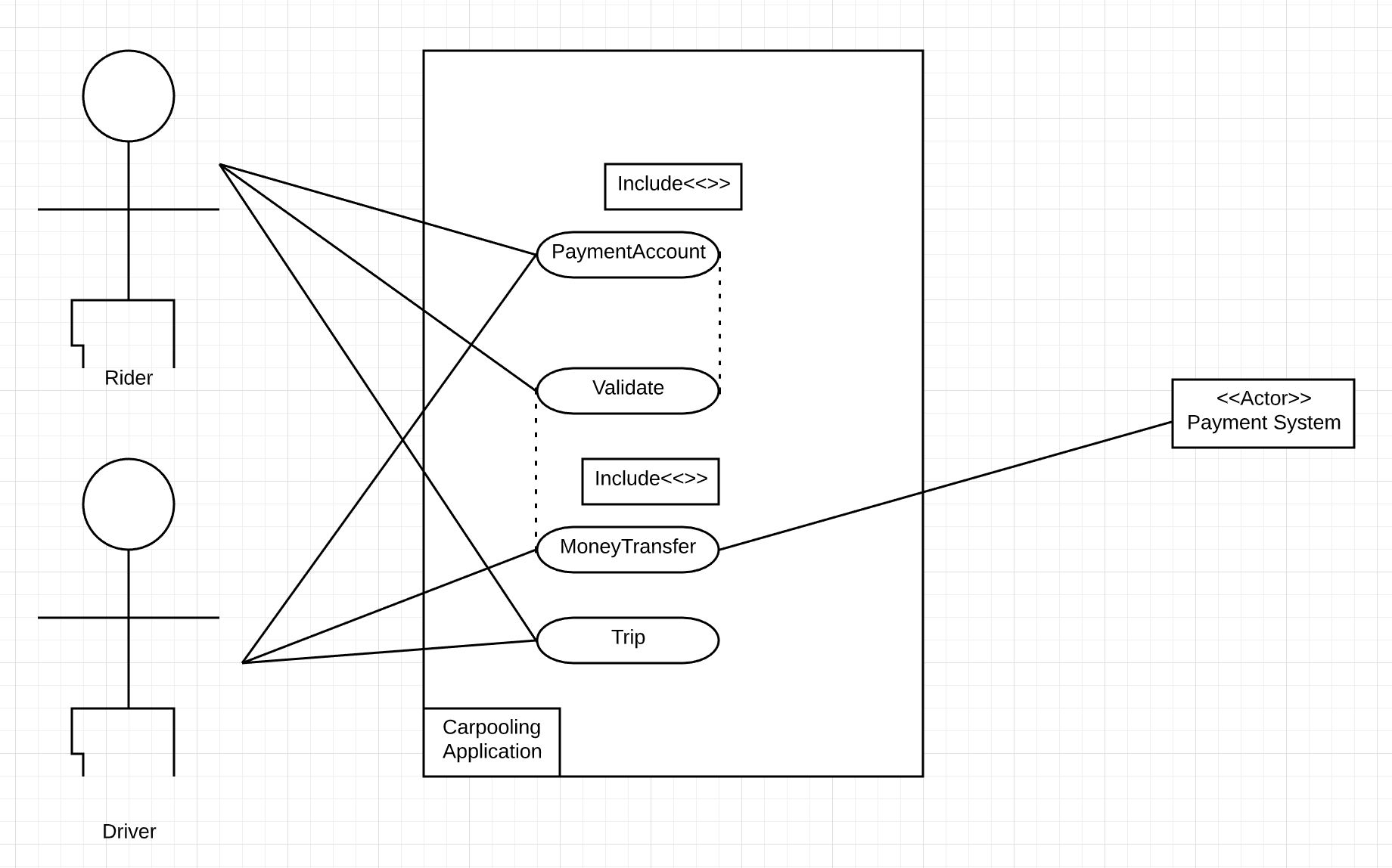
Access to check the balances of the payment accounts

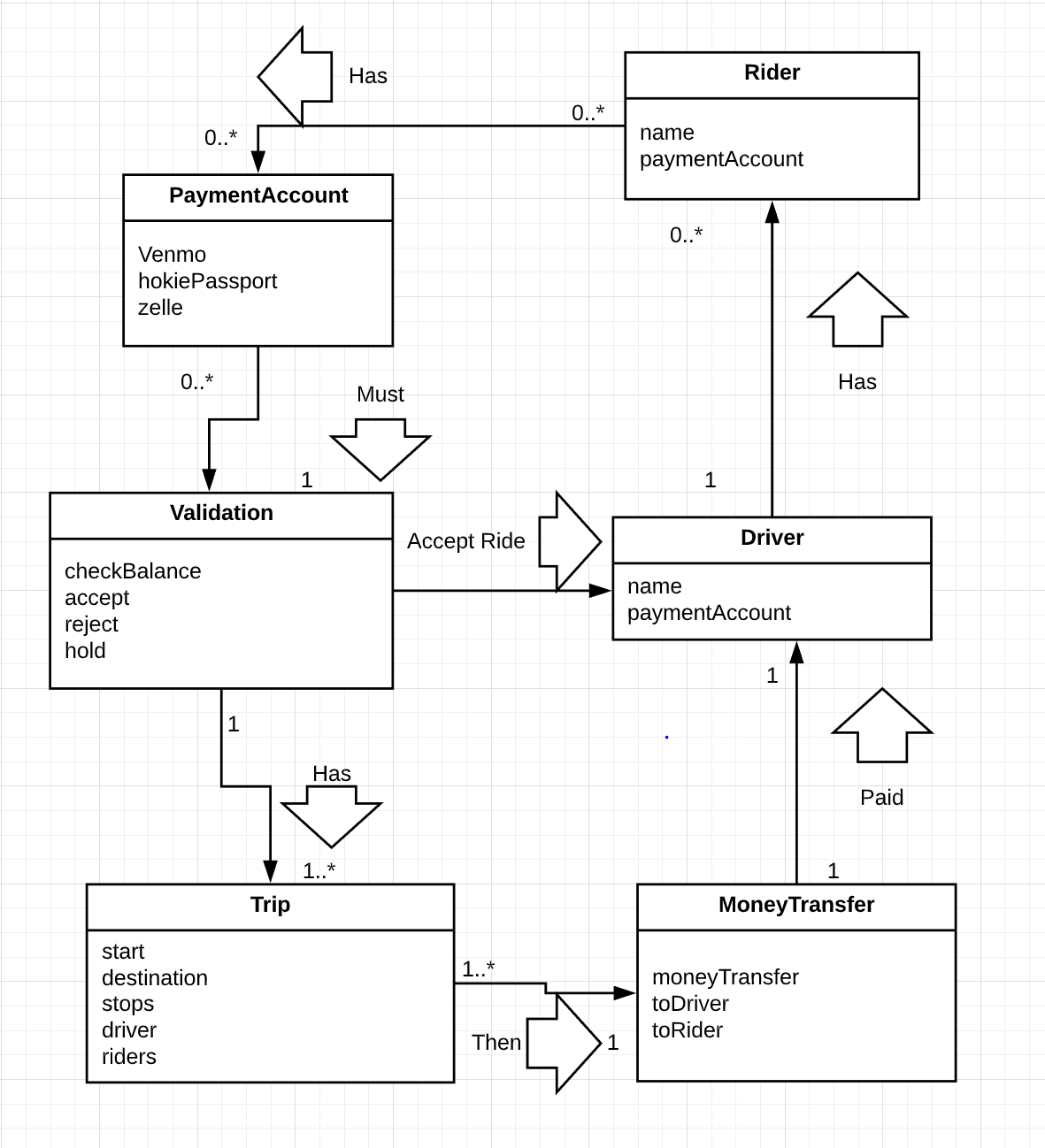
**Technology and Data Variation List:**

8a. Technology to check the balances

**Frequency of Occurrence:**

Everytime a rider requests a ride

**Use Case Diagram and Conceptual Class Diagram for Use Cases 14 - 16**



**Supplementary Specifications**

Screen Adaptation for different types of devices and systems

Availability on a common plane to access our application (Apple’s App Store, Google's Play Store)

Reliability of requesting rides and sending information (Transaction Summary, email confirmation)

Log failures

System response within 30 seconds