Requirements Analysis

Fully Dressed Use Cases:

Manage Resources

ID: UC-01

Primary Actor: Player

Stakeholders and Interests: Player: wants accurate update of resource stats.

Goal: To manage the virtual student's resources such as food, hygiene, and sleep.

Preconditions: The player is logged into the game.

Success Guarantee: The resource stats are updated for the player based on the player's actions.

Main Success Scenario:

1. The player does an action that changes their resources (eat, buy food, sleep, shower).
2. The game updates these resource values.
3. The game provides feedback on the student's resource status.

Extensions:

2a. The updated resource value exceeds the game's maximum value for the resource:

1. The game updates the resource value or level to the maximum value
2. The game notifies the player that they have reached the maximum resource value

Technology and data variation list:

1a. The resource value will be a quantifiable description of a student's level in that resource, expressed as a percentage.

Customize Character

ID: UC-02

Primary Actor: Player

Stakeholders and Interests: Player: wants ability to customize appearance and environment of virtual student.

Goal: To customize the appearance and environment of the virtual student.

Preconditions: The player is logged into the game.

Success Guarantee: The character and environment are updated based on the player's choices.

Main Success Scenario:

1. The player selects a customization option.
2. The player chooses a new appearance option for their character (new shirt, haircut, etc…).
3. The player selects new environment settings.
4. The game updates character and environment.

Extensions:

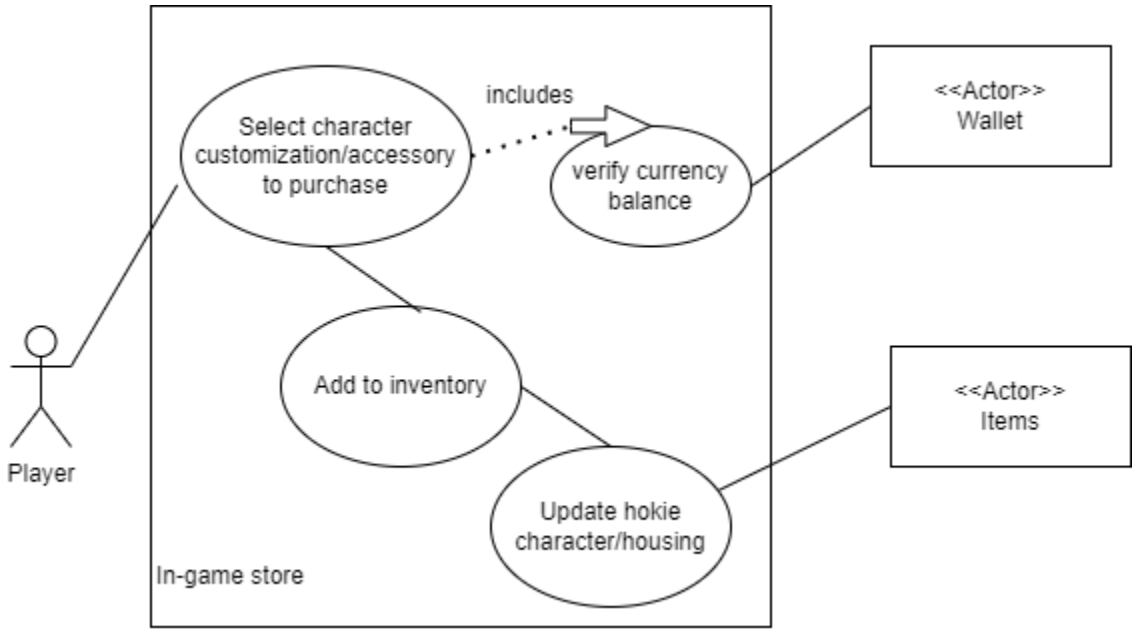
1a. The selected customization option is not available.

1. The game displays a message to the player and rejects the selected option.
2. The player can select a customization option.
3. Repeat from step 1 if option is not available.

2a. The selected appearance option is not available.

1. The game displays a message to the player and rejects the selected option.
2. The player can select an appearance option.
3. Repeat from step 1 if option is not available.

Technology and data variation list:



Check balance

ID: UC-03

Primary Actor: Player

Stakeholders and Interests: Player wants to check their current wallet balance.

Goal: Provide the player with their current wallet balance.

Precondition: The player is logged into their account.

Success Guarantee: The the currency amount currently in the player's wallet is displayed to them.

Main Success Scenario:

1. The player navigates to account balance or the item shop (where the balance will be shown in the corner).
2. The correct account balance is displayed.

Technology and data variant:

2a. The balance is stored and displayed as a positive integer.

Purchase Item

ID: UC-04

Primary Actor: Player

Stakeholders and Interests: Player wants ability to purchase items from item store. Item store wants to sell items to player.

Goal: To enable player to purchase items using in-game currency.

Precondition: The player is logged into the game.

Success Guarantee: The player owns the purchased item and the total cost of the item is subtracted from the player in-game wallet.

Main Success Scenario:

1. The player navigates to the item shop.
2. The player selects an item to purchase.
3. The player confirms the purchase.
4. The player owns the item and the total cost of the item is subtracted from their in-game wallet.

Extensions:

2a. The player does not have enough currency to purchase the item.

1. The game displays a message saying that the player does not have enough currency, rejecting the selected option.
2. The player gets placed back into the item shop to select another item.
3. If the player selects another item they cannot purchase, repeat from step 1 until a successful purchase is made or the player leaves the item shop.

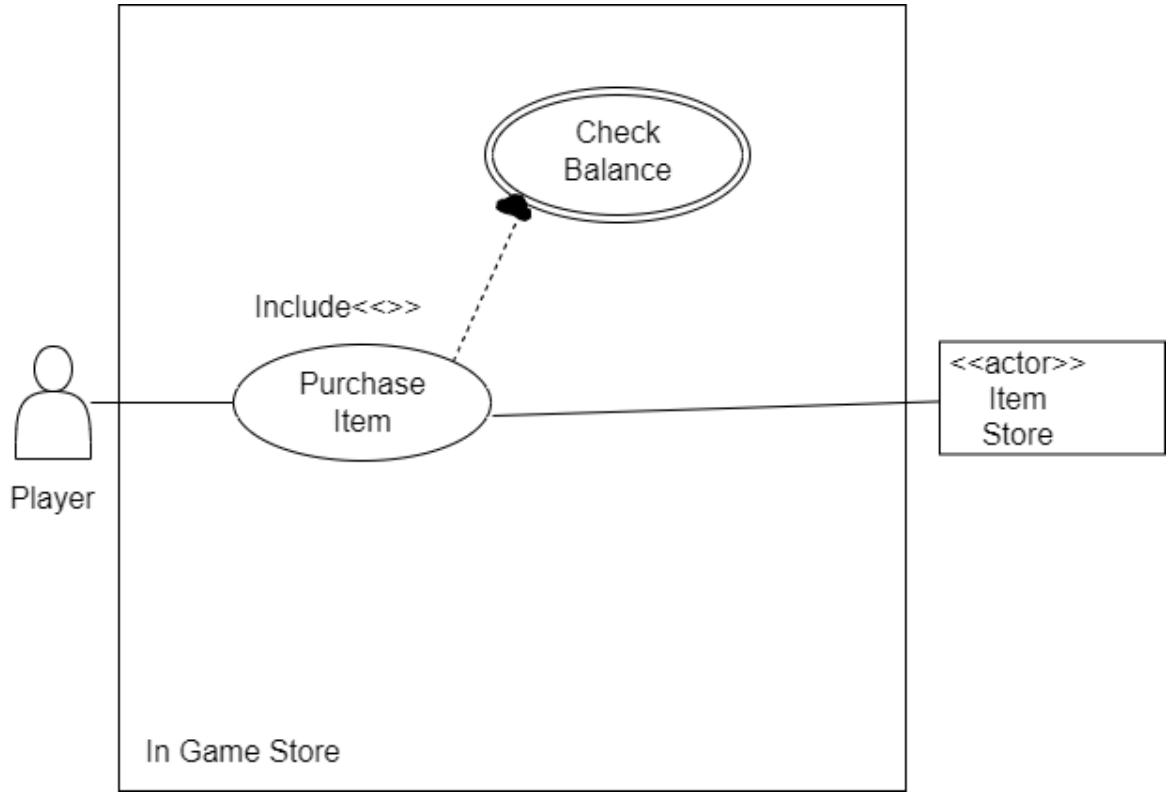
3a. The player does not confirm the purchase.

1. The purchase is canceled and the player is back in the item shop.

Technology and data variation list:

2a. The items are identified by unique integer identifiers.

Use case diagram for Purchase Item and Check Balance:



Check In

ID: UC-05

Primary Actor: Player

Stakeholders and Interests: Player wants to log in to the game to collect their daily rewards. The game developers want to reward the player for logging in to retain users.

Goal: To give players rewards for logging in daily.

Precondition: The user has an account to play the game.

Success Guarantee: The user keeps the rewards after logging in.

Main Success Scenario:

1. The player logs into the game.
2. The player reaps the daily rewards for logging in.

Extensions:

1a. The player does not log in on a day.

1. The player does not obtain the daily login rewards.
2. The daily reward streak is reset the next time the player logs in.

Add to wallet

ID: UC-06

Primary Actor: Player

Stakeholders and Interests: Player: the player wants currency added to their wallet after completing activities that reward with currency.

Goal: To add currency to the player wallet.

Precondition: The player is logged into their account.

Success Guarantee: The currency amount awarded by check-in or daily activities is added to the player wallet.

Main Success Scenario:

1. The player completes an activity or a daily log-in that awards with currency.
2. The currency amount is added to the player's wallet.

Extensions:

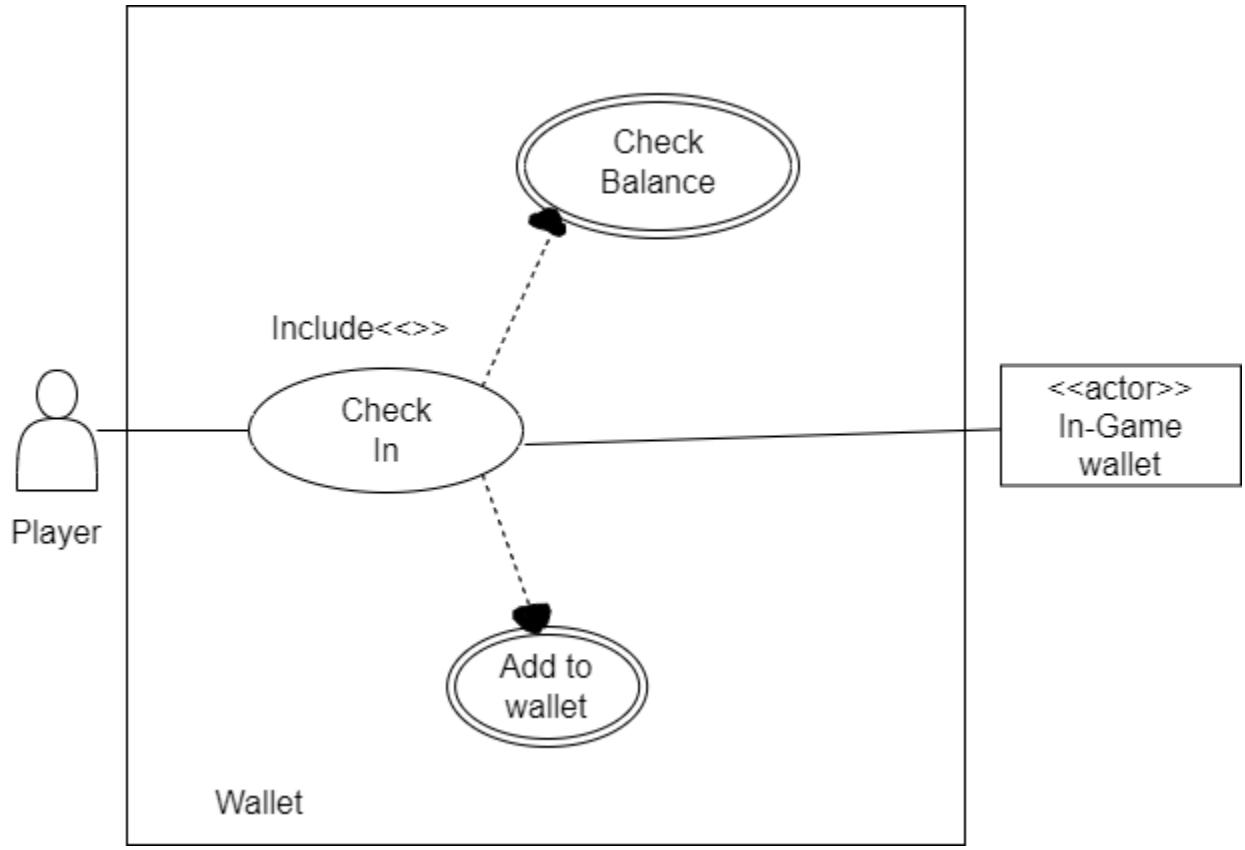
1a. The user does not complete the activity.

1. The award amount is not added to the wallet.

Technology and data variant:

2a. The currency is expressed as a positive integer data value.

Use case diagram for Check In, Check Balance, and Add to wallet:



Participation in multiplayer activities

ID: UC-07

Primary Actor: Player

Stakeholders and Interests: Player: The player wants to engage with other players in multiplayer activities for rewards and progression.

Goal: Allow players to participate in multiplayer tasks, like group study sessions or campus events to promote social interaction.

Precondition: The player is logged into the game and has access to multiplayer features.

Success Guarantee: The player participates in multiplayer activities and gets rewards and progression points.

Main Success Scenario:

1. The player joins a multiplayer event or task in the game.
2. The game matches the player with other players doing the same activity.
3. Players collaborate or compete against each other in that event.
4. The performance of the player is tracked and rewards are given accordingly.

Extensions:

2a. If there are not enough players online in the game, the game notifies the player and gives the player an option to play solo.

Tracking academic progress

ID: UC-08

Primary Actor: Player

Stakeholders and Interests: Player: The player wants to track their academic progress in terms of grades and overall GPA.

Goal: The goal is to provide a way for the player to monitor their academic progress and identify areas of improvement.

Precondition: The player is logged into the game and enrolled in classes for the semester.

Success Guarantee: The player can see their academic progress, including grades and GPA, on a visually appealing dashboard with charts and graphs.

Main Success Scenario:

1. The player navigates to the academic progress screen.
2. The game calculates changes in academic performance based on attendance, study routine, and exam results.
3. The updated academic information is displayed to the player.

Extensions:

2a. If the player has missed an exam, the game notifies the player and reflects the change in academic progress accordingly.

Time management system

ID: UC-09

Primary Actor: Player

Stakeholders and Interests: Player: The player wants to manage daily activities such as attending classes, studying, working out, and maintaining personal needs.

Goal: Allow players to manage their time by balancing activities like attending classes, studying, sleeping, and eating.

Success Guarantee: The in-game schedule of the player is updated based on their decisions and actions.

Main Success Scenario:

1. The player schedules daily activities like classes, studying, and meals.
2. The game tracks the player’s time spent studying and adjusts the player's exam readiness accordingly.

Extensions:
2a. If the player skips a scheduled study session, the game notifies the player and reduces exam readiness.

Special Events and Real Time Dependencies

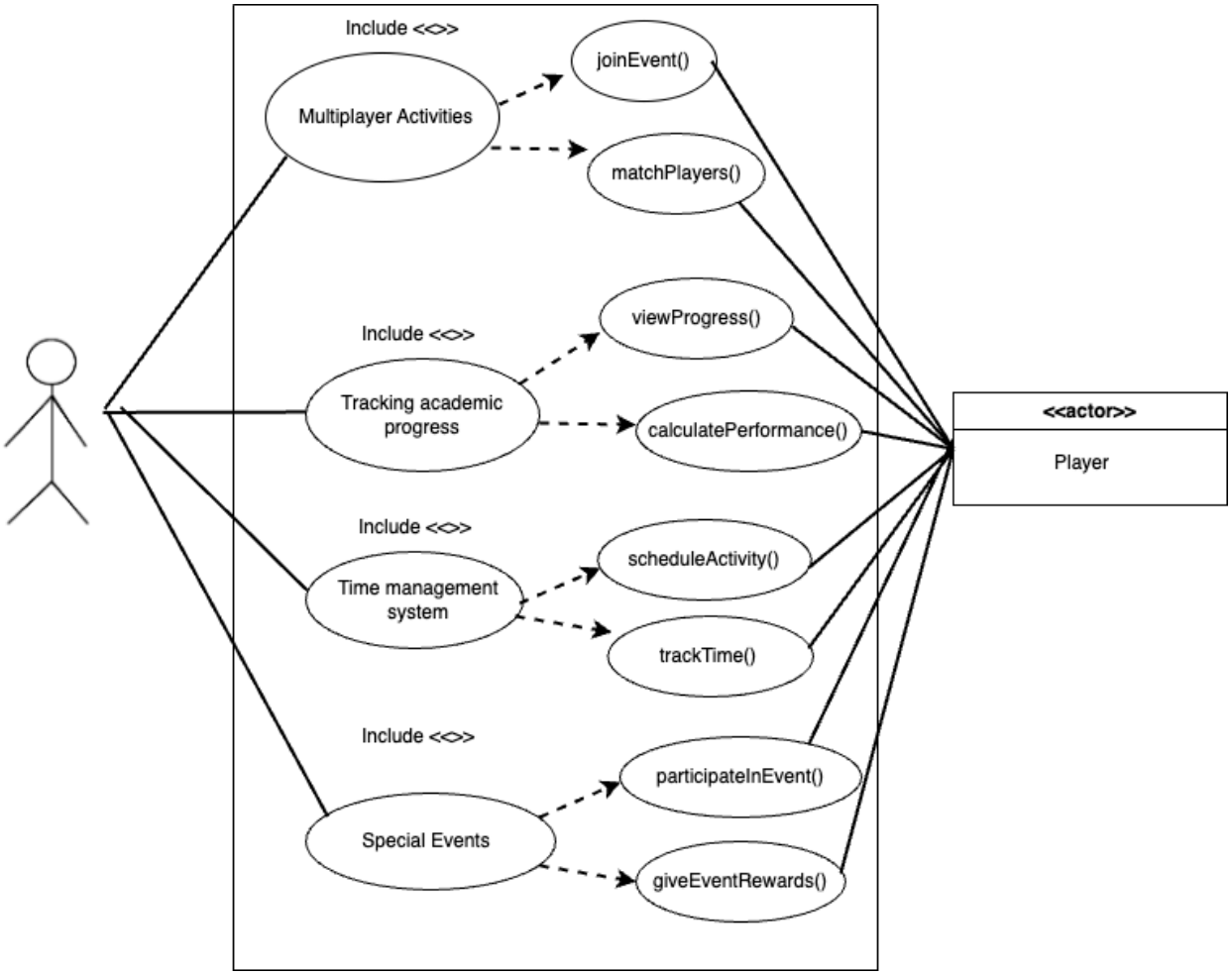
ID: UC-10
Primary Actor: Player
Stakeholders and Interests:

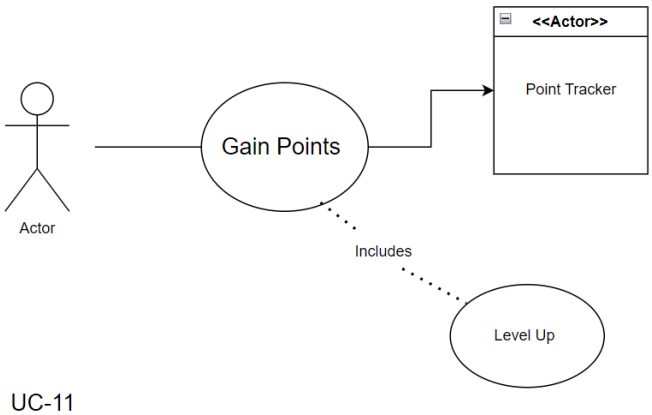
1. Player: The player wants to participate in special in game events based on real time calendar events (eg holidays, exams, etc) for exclusive rewards and progression points.
2. Game Developer: The developer wants to increase player engagement through special rewards and progression points.

Goal: Allow players to engage in special events tied to events, such as holidays, exams, or campus events, for unique rewards and experiences.
Precondition: The player is logged into the game and a special event is active based on the in game calendar.
Success Guarantee: The player successfully participates in a special event and earns rewards or bonuses associated with the event.
Main Success Scenario:

1. The game displays a notification about the event with potential rewards.
2. The player logs into the game during a special event.
3. Upon successful participation, the player receives exclusive rewards or bonuses.
4. The event concludes, and the player's progress and rewards are saved in the game.

Extensions:
2a. If the player logs in outside the event window, they receive a notification that no event is currently active.
4a. If the player fails to complete the event challenges, the game records their progress but withholds rewards until all tasks are completed.





UC-11

Tracking Experience/ Leveling Up

ID: UC-11

Primary Actor: Player

Stakeholders and Interest:

- Player: The player can level-up their Hokie, effectively giving the user a feel of success and progression

Goal: For users to enjoy upgrading their Hokie and continue playing based on the progression and competitive aspect.

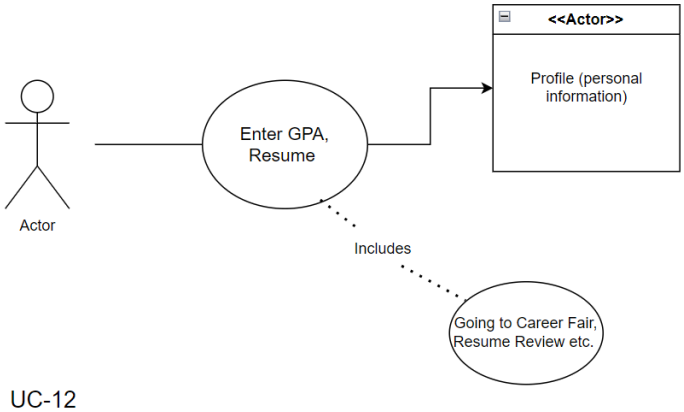
Success Guarantee: Experience and score are accumulated through various activities.

Main Success Scenario:

1. User plays mini games and earns points.
2. User takes care of Hokie, completes multiple tasks throughout the day (the more efficiently you work the better score you get).

Extensions:

1. Player does not play any mini-games, they do not gain points or XP
2. Player does not take care of Hokie, player loses points and Hokie's XP is stagnant
3. Player loses mini-games and doesn't gain any points



UC-12

GPA, Resume and Network Score

ID: UC-12

Primary Actor: Player

Stakeholders and Interest:

- Player: User uses their real-life GPA and resume in order to boost their XP. This can be an incentive for the player to do well in their real-world college life.

Goal: To encourage users to do well in boosting GPA and creating a network/ resume

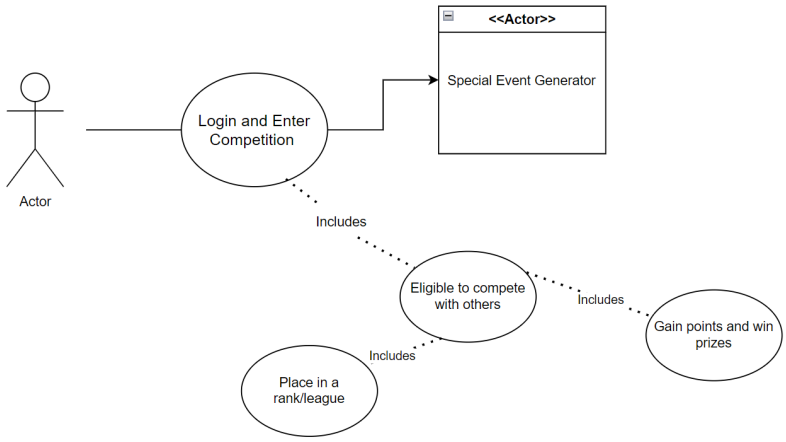
Success Guarantee: The player successfully participates in a special event and earns rewards and bonuses associated with the event.

Main Success Scenario:

1. The game displays an event notification with potential rewards.
2. The player logs into the game during a special event.

Extensions:

1. The player does not participate during a special event and does not gain experience in the game (such as a career fair or resume review)
2. The player does not put in their resume and GPA, they do not receive any benefits.



UC-13

Rankings and Competition Scores

ID: UC-13

Primary Actor: Players

Stakeholders and Interest:

- Player: Players can compete against other Hokies in order to win prizes and potentially more XP than you would in mini-games. The player would then be ranked and put into leagues based on their competitive scores and/or XP

Goal: To allow players to earn more points during special events and create a friendly but competitive environment for other players to interact.

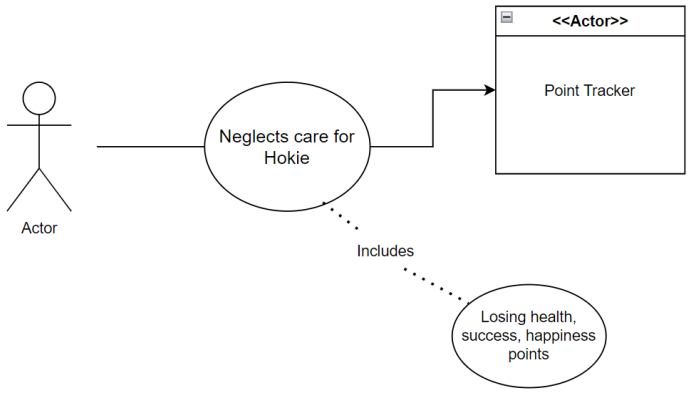
Success Guarantee: The player enters into a competition and places at a high ranking, they earn prizes and XP depending on their status which will help them level up.

Main Success Scenario:

1. Player enters a competition and wins games
2. The player earns points and a rank in a league

Extensions:

1. Player does not win any prizes and loses games → user will rank low and earn minimal XP
2. Player enters competition but does not play any games → user will rank low and earn minimal XP if any at all



UC-14

Loss of Experience/ Score

ID: UC-14

Primary Actor: Player

Stakeholders and Interest:

- Player: The player can lose points and experience if they do not take care of their Hokie or miss important events that they signed up for

Goal: To incentivize players to continue playing and leveling up their Hokie otherwise they can lose their rank and points.

Success Guarantee: If the player misses daily tasks such as homework, tests/quizzes, and self-care then the player will lose points and potentially level down

Main Success Scenario:

1. Player stops playing for a couple of days and the Hokie becomes sick, grades lower due to missing homework and attendance due to not going to class. Players will lose health, happiness and success points.

Extensions:

1. Player misses tests but comes back and tries to re-boost their status, player will win back any potentially lost points, including health, happiness and academic success

Event Trigger System for Special Events

ID: UC-15

Primary Actor: Player

Stakeholders and Interest:

- Player: The player wants to participate in special events for rewards
- Game Developer: Increases engagement by adding exclusive in-game events

Goal: Automatically trigger in-game special events based on real-world or in-game dates

Preconditions: The player is logged into the game and a scheduled event is active

Success Guarantee: The event is triggered at the correct time and players can participate in the event

Main Success Scenario:

1. The game system checks the event schedule based on real-time calendar integration or in-game date progression
2. A special event is triggered at the designated time
3. The player is notified that a special event is active
4. The event-related features are made available

Extensions:

1a. If the game is offline during the scheduled event time, the event is triggered the next time the player logs in

Ab. If the event schedule changes, the game updates the trigger system and notifies players of the change

Technology and Data Variation List:

Event triggers are based on real-time API integration for the calendar events

Real-Time Scheduling and Event Notifications

ID: UC-16

Primary Actor: Player

Stakeholders and Interest:

- Player: The player wants to be notified of upcoming events in real-time
- Game Developer: Wants Players to be well-informed of special events to increase engagement

Goal: Notify players about upcoming or ongoing special events in real-time

Preconditions: The player is logged into the game and connected to notification system

Success Guarantee: Players receive timely notifications about event and can prepare or join

Main Success Scenario:

1. The game system identifies an upcoming special event based on real-time or in-game scheduling
2. The player is notified via in-game pop-up and/or push notification
3. The notification includes details about the event, rewards and participation instructions
4. The player acknowledges the notification and can opt to participate in the even

Extensions:

1a. If the player is offline, they receive the notification the next time they log in

1b. If notifications are disabled the player sees the event notification in the game's event section instead

Technology and Data Variation List:

Notifications can be sent via in-game message systems or mobile push notifications

Dynamic Event Participation

ID: UC-17

Primary Actor: Player

Stakeholders and Interest:

- Player: The player wants to dynamically join or leave special events in the game

- Game Developer: Wants Players to be actively engaged and able to join events at any point during the event window

Goal: Allow players to dynamically participate in events during their active period

Preconditions: The player is logged into the game and a special event is active

Success Guarantee: The player successfully joins the event and their participation is tracked

Main Success Scenario:

1. The player sees a notification or prompt about the special event
2. The player chooses to participate in the event
3. The game dynamically places the player into the event
4. The player's actions and progress during the event are tracked
5. The player's rewards and progression points are updated

Extensions:

2a. If the player attempts to join after the event has ended, they receive a notification that the event in no longer available

2b. If the player leaves the event midway, they may not receive full rewards

Technology and Data Variation List:

The event participation system uses real-time synchronization to track player actions and progress throughout the event

Event-Specific Goals or Tasks

ID: UC-18

Primary Actor: Player

Stakeholders and Interest:

- Player: The player wants to complete special event tasks to earn exclusive rewards
- Game Developer: Wants Players to engage deeply with special events through challenging tasks

Goal: Assign players event-specific tasks that are only available during the event period

Preconditions: The player is logged into the game and a special event is active

Success Guarantee: The player completes event-specific tasks during an event and earns corresponding rewards

Main Success Scenario:

1. The game assigns temporary tasks or goals that are specific to the special event
2. The player reviews the event-specific goals and begins working on them
3. The player completes the tasks within the event period
4. The player receives special rewards based on task completion

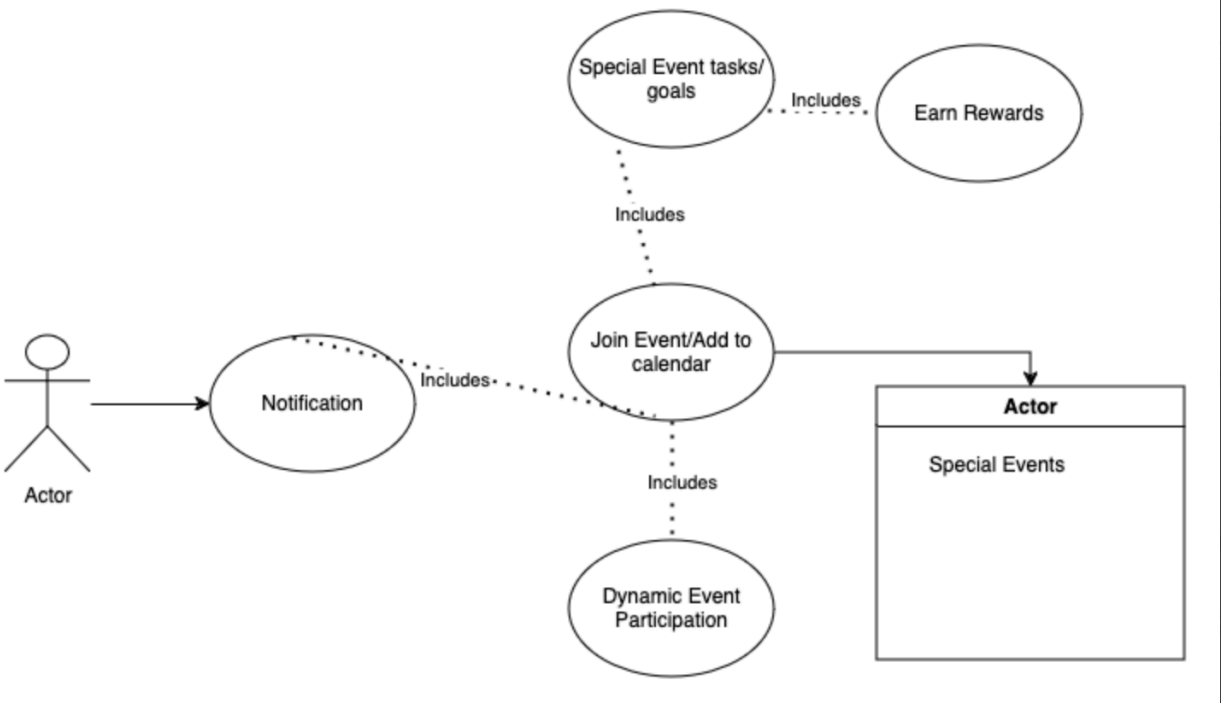
Extensions:

2a. If the player does not complete the tasks before the event ends, the tasks expire

Technology and Data Variation List:

Tasks and goals are dynamically generated for each event and linked to player

Diagram for UC-15, UC-16, UC-17, and UC-18:



Switch Housing

ID: UC-19

Primary Actor: Player

Stakeholders and Interests: Player: wants to change their housing (dorm, apartment, townhouse, house)

Preconditions: N/A

Success Guarantee: Background for housing screen is changed. Appropriate balance is deducted for player's wallet (currency field)

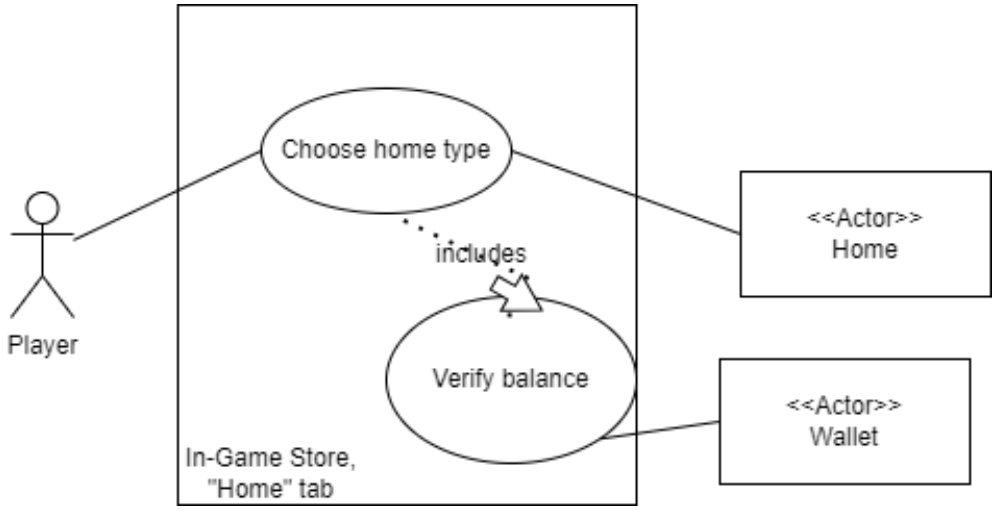
Main Success Scenario:

1. Player selects "Home" tab in the in-game store
2. Player can choose from 4 housing types (dorm, apartment, townhouse, house)
3. Player's selection updates the player's "housing style" attribute

Extensions:

2a. Player cannot afford to switch to their chosen home style; display popup saying "You don't have enough money yet!"

Technology and data variation list: N/A



Adopt a Pet

ID: UC-20

Primary Actor: Player

Stakeholders and Interests: Player: wants to adopt a pet

Preconditions:

Success Guarantee: The successfully adopted pet and its correct animal type/color are always displayed on the housing screen. Appropriate money is deducted from player wallet (currency field).

Main Success Scenario:

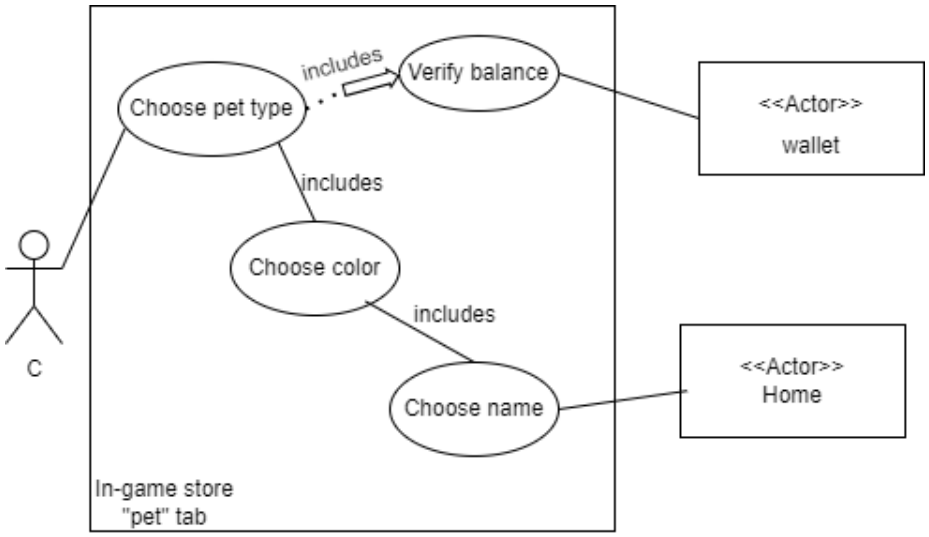
1. Player selects “Pet” tab in the store
2. System verifies that player stats will allow pet adoption
3. Player selects an animal type (cat, dog, bird)
4. Player selects a color
5. Player names the pet
6. Adoption status and pet attributes are linked to the player’s profile
7. Appropriate money is deducted from the player’s wallet

Extensions:

2a. Player lives in a dorm; system displays popup saying “You can’t adopt a pet while living in the dorms!”

2b. Player does not have enough money; system displays popup saying “You can’t afford a pet right now!”

Technology and data variation list: N/A



Gacha Game Singular Spin

ID: UC-21

Primary Actor: Player

Stakeholders and Interests: Player: wants to play gacha minigame and spin for a random personal/housing accessory.

Preconditions:

Success Guarantee: The randomized personal/housing accessory is added to the player's inventory and the appropriate number of tickets is deducted from the player's wallet (ticket field).

Main Success Scenario:

1. Player selects mini-game menu
2. Player selects gacha game
3. Player clicks the "Singular Spin" button
4. System validates the player can afford to spin
5. System selects a randomized personal/housing accessory
6. The accessory is added to the player's inventory
7. Appropriate ticket number is deducted from player's wallet (ticket field)

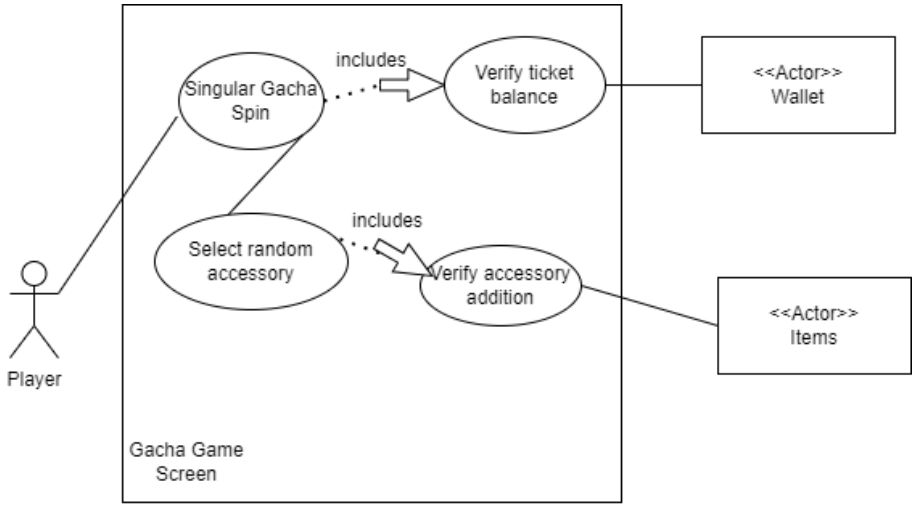
Extensions:

3a. Player cannot afford to play the minigame; system displays popup saying "Come back to spin when you have more tickets!"

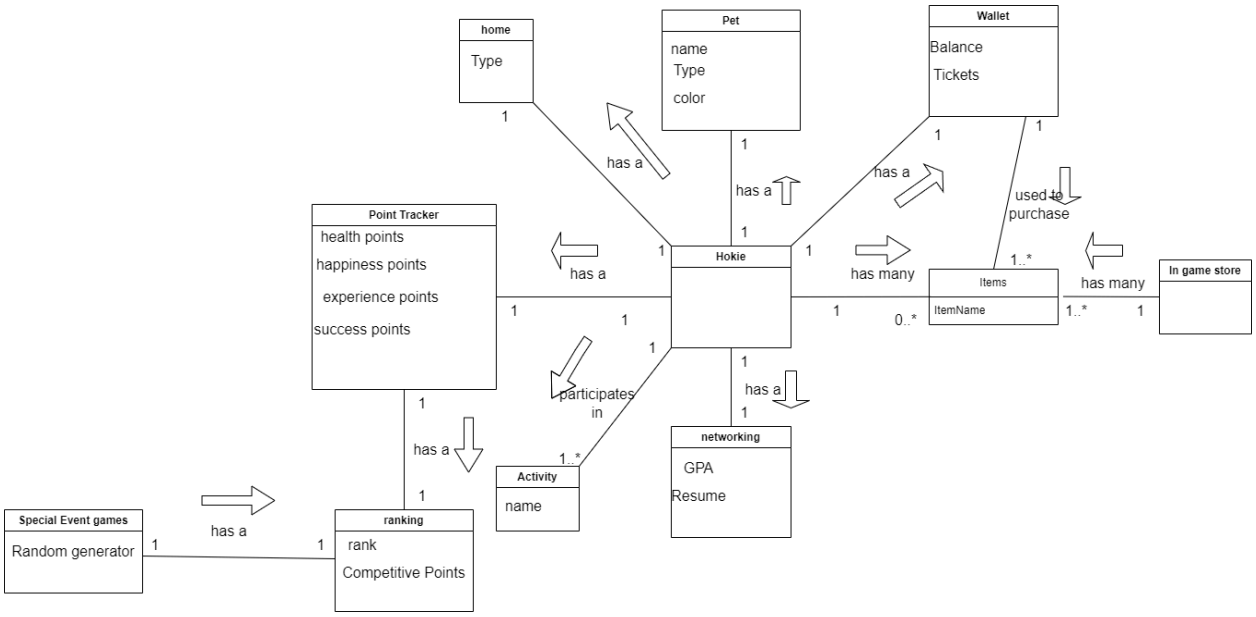
5a. System's randomly chosen accessory is invalid

1. Player already owns all possible personal/housing accessories; system displays popup saying "You have all accessories already!"
2. Player already owns the randomly chosen accessory; system re-chooses

Technology and data variation list: N/A



UML Diagram:



Supplementary specifications (Non functional requirements)

1. Score tracker should be able to update within seconds of calculation
2. The game must be compatible with multiple platforms, including iOS, Android, and desktop browsers, ensuring a consistent user experience across all devices.
3. Icons will be deliberately chosen so that users can intuitively navigate the game.
4. Organized software development system/workflow with regular upkeep/documentation for debugging ease
5. Sensitive data, like login info, must be securely encrypted.

6. Handle a database of ~ 500 users that allows multiplayer features (ranking, categorization, and a friends list)