


[PAR-171] [Parking] Driver parking web app (GENERIC) Parking EZ Pro Created: 10/14/24 Updated: 10/21/24 Resolved: 10/14/24

Status:	New		
Project:	<a href="#">Cubework Parking</a>		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Story	Priority:	Major
Reporter:	<a href="#">Bassel Matta</a>	Assignee:	<a href="#">zhenyan.guo</a>
Resolution:	Unresolved	Votes:	0
Labels:	Parking		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Attachments:	 <a href="#">parking ez pro .pdf</a>		
Dev Hours:	0		
Story Points:	0		
Test Hours:	0		
Test Points:	0		

Description

Parking EZ Pro

Workflow:

Overview

This workflow is designed for a parking solution that integrates both hardware (gate barriers, cameras, QR codes, signs) and software (parking app, backend system) to provide an automated parking experience for drivers and parking lot owners. The system offers flexible check-in and check-out processes, accurate license plate recognition, and a seamless payment experience, with a key focus on time tracking and billing.

Hardware Components

- Gate Barrier:** Controls entry and exit of the parking lot.
- Cameras:** Positioned at the gate, used for automatic license plate recognition.
- QR Codes:** Displayed at the entrance and exit points to trigger key actions.
- Signs:** Direct drivers on how to use the parking app and hardware.
- QR Scanner:** To scan the QR code of the user after payment to allow them to exit the gate.
- Speaker:** To let the user know of any issues and direct them to the right way.
- LED Screen:** For showcasing the number of spots available in the lot.

Check-in Process (Driver Side)

- Arriving at the Parking Lot**
  - The driver pulls up to the gate barrier.
  - The driver scans the **QR code** posted on the sign using their phone.
- Trigger License Plate Recognition:**
  - The QR code scan triggers the **camera** to take a picture of the car's license plate.
  - The system records the **license plate number**, along with the **time** and **date** of the check-in.
- Handling Multiple Entries:**
  - The system will account for multiple check-ins per day, ensuring each check-in and outbound event is properly tracked.
  - If the same vehicle checks in multiple times, the system associates each check-in with a specific event, even if different user accounts are used.
  - Each entry will trigger a new session for billing purposes, using the correct check-in time for calculation.
- Time Tracking and Rate Calculation:**
  - Once the license plate is captured and the check-in is logged, the system will start calculating the time spent in the lot.
  - The rate per hour is determined by the parking lot owner's rate settings, which are linked to the specific parking lot (via the scanned QR code).

Check-out Process (Driver Side)

- Payment Initiation via Kiosk**
  - The driver approaches the exit and heads to a **kiosk** located nearby.
  - They can scan the provided **QR code** with their phone to trigger the payment interface.
  - Alternatively, they can manually enter their **license plate number** to retrieve their latest check-in information.
- Price Calculation:**
  - The system checks the latest **check-in time** for that license plate, calculates the total time spent in the parking lot, and determines the cost based on the rate associated with that parking lot.
  - The driver is then prompted to make the payment.
- Payment Process:**
  - The driver makes the payment via their phone or through **Apple Pay**/credit card options provided on the kiosk.
- Grace Period:**
  - After payment, the driver is given a **15-minute grace period** to reach their car and exit the parking lot. This grace period is added to accommodate potential delays, such as traffic within the lot.
- Exit Process:**
  - Upon reaching the exit, the **camera** reads the license plate again.
  - If the system detects that the car has paid within the last 15 minutes, the **gate automatically opens** for the driver.

Error Handling (License Plate Recognition Issues)

- Unrecognized License Plate at Exit**
  - If the camera fails to recognize the license plate, the driver can scan the QR code at the gate and manually enter their **license plate number**.
  - The system will check if payment has been made within the last 15 minutes. If payment is confirmed, the driver can click an "open gate" button in the app, and the gate will open.
- No Prior Payment Detected:**
  - If no payment is detected, the driver will be prompted with a message stating that payment is required.
  - The driver can then make the payment on the spot using **Apple Pay** or by entering their credit card information.
  - Once payment is processed, the gate will open, allowing the driver to exit.

Developer Considerations

- Multiple Entry/Exit Handling:**
  - The system should maintain a robust log for every check-in and check-out, ensuring that even if the same car re-enters multiple times, the billing and session tracking are handled correctly.
  - Care should be taken to handle cases where different accounts might be used for the same car, with clear mappings to ensure accurate billing.
- Real-Time License Plate Recognition:**
  - The camera system must work seamlessly with the backend to recognize the license plate and associate it with the correct check-in time.
  - Ensure fast and reliable communication between the QR scan, camera trigger, and database logging for accurate records.
- Accurate Billing System:**
  - The backend must ensure that the billing calculation is based on the actual check-in and check-out times, with proper handling of rounding and fractional hours (e.g., charge per hour or partial hours).
  - Ensure that each QR code is tied to a specific parking lot to apply the correct rates.
- Grace Period Logic:**
  - Implement logic that enforces the 15-minute grace period for checkout, ensuring that the car can exit without additional charges within this time frame.
- Fallback for License Plate Recognition Failures:**
  - Implement a smooth fallback process where users can enter their license plate number manually and still proceed to payment or exit if the camera fails.
- Payment Gateway Integration:**
  - Ensure secure and seamless integration with payment systems (Apple Pay, credit cards) to provide fast, reliable, and secure transactions.

This architecture should ensure a smooth and efficient parking experience, while also handling various edge cases and maintaining accuracy in time and billing calculations.

Comments

Comment by Bassel Matta [ 10/14/24 ]

Original Notes: (ignore)

this is going to be the workflow for the generic parking app for the driver for phase 2.

this solution we will offer to anyone who want to use our parking app. The solution comes with the equipment so the owner of the parking lot can buy it from us. They can buy the gate barrier, camera, QR codes and signs.

we will allow the driver to do the following for check-in process:

- come to gate barrier
- scan the qr code from the sign with their phone
- this will trigger the camera to read and record the license plate number and save it in the system and save the time and date in which this car scanned the barcode and came to the gate...keep in consideration if the same car did an outbound and came back in to check-in again for round 2 and make sure to do calculation based on the correct check-in time and date in case the same car comes in multiple times per day or uses different accounts each time, system should allow that and accurately identify which account and which check-in this is.
- start counting the time for this car to calculate how much we will charge them based on the rate per hour that the seller has in their account for this specific parking lot based on the barcode scanned which is associated with a specific parking lot.

- For the checkout process, we will have a kiosk that will allow the user to scan with their phone and make the payment on the phone based on the time they have stayed we will calculate the payment.
- the user will enter the license plate number, so the system will try to find this license plate latest entry in the system and will make calculation for price and allow the driver to make payment.
- then we will give the driver 15 minute grace period to go to their car and exit the parking lot. the 15 minute is to allow the driver to get to their car and allow of any parking lot traffic jam.
- then when they go to the gate within this 15 minutes after the payment, the gate will open when the camera reads the license plate and see that this car has paid within the past 15 minutes.
- what if the camera doesn't read the correct license plate? allow the user to scan qr code by the gate and click on the exit gate and enter the license plate number to get identified, then they click open gate, if they have made the payment full within the past 15 minutes then the gate will open. if they have not made the payment then we will prompt them a message saying they need to make payment then they can use apple pay or enter credit card info on the spot and make payment then gate will open.

Comment by Bassel Matta [ 10/14/24 ]

EDGE CASES:

When developing the parking app, it's crucial to consider and handle various edge cases to ensure that the system works seamlessly under all conditions. Below are some potential edge cases that need to be addressed, along with strategies to handle them:

1. Camera Fails to Recognize License Plate (At Check-in or Check-out)

- **Scenario:** The camera might not correctly read the license plate due to poor lighting, dirty plates, or camera malfunction.
- **Handling:**
  - Provide a **manual override** option at the gate, where the driver can scan the **QR code** and manually enter the license plate number.
  - After manual entry, the system checks the most recent record of the license plate and processes the action (check-in, check-out) accordingly.
  - Implement a fallback to alert the parking lot staff in case of repeated failures with license plate recognition for maintenance or cleaning of the cameras.

2. Multiple Check-ins in the Same Day

- **Scenario:** A car may enter and exit the same parking lot multiple times a day, potentially using different user accounts.
- **Handling:**
  - Store **unique session IDs** for every entry/exit event, including timestamps.
  - Allow the system to handle multiple sessions per vehicle. Ensure that each check-in and check-out is independently tracked, even if the same license plate is detected more than once per day.
  - Differentiate users by **account association**, even if they use the same car multiple times.
  - Ensure that billing is applied based on **each unique session** using the correct check-in time for that session.

3. Vehicle Check-in Without Check-out (Or Vice Versa)

- **Scenario:** A car checks in but never checks out, or a car checks out without a recorded check-in (e.g., system glitch or user error).
- **Handling:**
  - **Timeout-based handling:** If a car checks in and no check-out event is recorded after a reasonable amount of time (e.g., 24 hours), automatically flag the session for review and potentially apply a flat rate.
  - In case a check-out is detected without a corresponding check-in (e.g., if a camera misreads a license plate), allow the system to alert the driver that they need to check-in and backdate the event if needed. This could be manually confirmed via app notification or kiosk interaction.

4. Lost Ticket or Payment Issues (Unable to Retrieve License Plate Data)

- **Scenario:** The system fails to retrieve the license plate record at checkout, or the user cannot remember their plate number.
- **Handling:**
  - Provide an option for the user to retrieve the license plate record by **account association**, such as entering their email or phone number, or by scanning the QR code associated with their check-in.
  - If no record can be found, allow the user to contact parking lot support for assistance.

5. Payment System Fails or Connectivity Issues

- **Scenario:** The user tries to make a payment, but the payment gateway is down or there's a connectivity issue.
- **Handling:**
  - Offer **multiple payment methods** (e.g., credit card, Apple Pay, or direct bank transfers) to provide redundancy in case one method fails.
  - Store payment information locally (temporarily) and retry processing the payment later if there is a connectivity issue, informing the user that they can proceed but the payment will be processed once the connection is restored.
  - Provide a **contactless payment method** as a backup at the kiosk (such as a physical card reader) in case the app cannot process payments.
  - Enable an **offline mode** where payment can be processed once the connection is re-established. Keep track of the time to ensure correct billing even in case of a delayed payment.

6. Driver Exceeds Grace Period (Post Payment)

- **Scenario:** A driver makes the payment but exceeds the 15-minute grace period before reaching the exit gate.
- **Handling:**
  - Notify the driver via the app when the 15-minute grace period is about to expire (e.g., at 10 minutes and 14 minutes).
  - If the driver exceeds the grace period, the system should recalculate the total parking fee and prompt the driver to pay any additional charges for the extra time before allowing the gate to open.

7. Wrong Car at Checkout

- **Scenario:** A different car tries to check out using a license plate number that doesn't match the entry event (e.g., a user enters the wrong license plate at the kiosk).
- **Handling:**
  - The system should verify the license plate by cross-referencing the **entry event** and **session information**.
  - If a mismatch occurs, prompt the user to **re-enter the correct license plate** or contact support for verification.
  - Display a clear error message at the kiosk or in the app, preventing incorrect check-outs and ensuring that the right vehicle is billed.

8. Invalid QR Code Scanning

- **Scenario:** A driver scans the wrong QR code (from another parking lot or outdated signage).
- **Handling:**
  - Implement **validation logic** to ensure that the scanned QR code is valid for the specific parking lot they are in, preventing check-ins or check-outs with invalid codes.
  - Provide a user-friendly error message explaining the issue and instructing the driver on what to do next (e.g., re-scan the correct code or manually enter the plate number).

9. Driver Forgetting to Check Out

- **Scenario:** The driver leaves the parking lot without going through the formal check-out process.
- **Handling:**
  - Implement **automatic check-out** by detecting when the vehicle leaves through the gate (using the camera) if the driver fails to check out manually.
  - Send a reminder via the app or email, prompting the user to complete the payment even after they have left the lot.
  - After a reasonable grace period, automatically apply any outstanding charges and bill the user's account.

10. System Downtime or Maintenance

- **Scenario:** The system (either the app or backend) goes down for maintenance or crashes.
- **Handling:**
  - Develop a **fail-safe mode** where drivers can still access the parking lot during system downtime. For example, the gate could open automatically after a set period if no license plate is recognized.
  - Provide an option for users to **prepay** for parking during maintenance or outages, allowing them to bypass the system temporarily.
  - Alert users in advance when scheduled maintenance will occur, or notify them of system outages in real-time through the app.
  - Implement a robust **logging system** to ensure all data (entry, exit, payments) is captured and synchronized once the system is back online.

11. Handling Fraudulent Activity

- **Scenario:** Users attempt to manipulate the system (e.g., by using false license plates or scanning a QR code multiple times without check-out).
- **Handling:**
  - Implement **license plate verification** using existing parking history and patterns to detect suspicious activity.
  - Flag accounts or sessions that exhibit unusual behavior, such as entering and leaving without proper payment or attempting to use fake QR codes.
  - Notify the parking lot owner and driver, and require additional verification (e.g., manual review or contact support).

12. Data Synchronization Issues

- **Scenario:** Data between the kiosk, app, and backend system isn't synchronized correctly, causing incorrect billing or check-in/out errors.

..

..

- **Handling:**
  - Implement real-time **data synchronization** across all systems with regular backups to avoid data loss.
  - Provide redundancy measures to ensure that if one component fails (e.g., kiosk), the app can handle check-out and payment processes.

Conclusion

By carefully considering these edge cases and implementing robust fallback mechanisms, the app will ensure a seamless user experience under various circumstances. Testing these scenarios thoroughly in different environments (poor lighting, different vehicles, etc.) will also help to ensure that the system works reliably in real-world conditions.

Comment by Bassel Matta [ 10/21/24 ]

UPDATED WORKFLOW BASED ON POWERPOINT:

Overview:

This workflow describes a parking solution that uses both hardware (gate barriers, cameras, QR codes, speakers, and LED displays) and software (parking app, backend system) to provide a seamless and automated parking experience. The focus is on efficient check-in/check-out processes, automatic license plate recognition, and easy payment solutions.

Hardware Components

1. **Gate Barrier:** Controls entry and exit of the parking lot.
2. **Cameras:** Positioned at the gate, used for automatic license plate recognition.
3. **QR Codes:** Displayed at the entrance and exit points to trigger key actions.
4. **Signs:** Direct drivers on how to use the parking app and hardware.
5. **QR Scanner:** to scan the QR code of the user after payment to allow them to exit the gate.
6. **Speaker:** To let the user know of any issues and direct them to the right way.
7. **LED Screen:** For showcasing the number of spots available in the lot.

Check-in Process (Driver Entry)

1. **Driver Scans QR Code:**
  - The driver scans the QR code displayed at the parking entrance using their smartphone.
1. **Phone Number Verification:**
  - After scanning the QR code, the driver is prompted to verify their phone number for further communication.
1. **Receive Verification Code:**
  - The driver enters the verification code sent to their phone, and a link to review their parking entry ticket is provided.
1. **License Plate Recognition:**
  - The system attempts to automatically recognize the license plate via the camera.
    - **If successful:** The recognized license plate is displayed for confirmation.
    - **If unsuccessful:** A speaker prompts the driver to manually input their license plate number.
1. **License Plate Confirmation:**
  - The system shows the recognized license plate. If it is correct, the driver confirms; if not, manual input is required.
1. **Gate Opens:**
  - Once the license plate is confirmed or entered, the gate barrier opens, allowing the driver to enter the parking lot.
  - Low Caps \* SYSTEM SHOULD RECORD THE TIME AND DATE OF THIS CAR AND START COUNTING TIME SO WE CAN KNOW HOW MUCH WE WILL CHARGE THEM.
1. **Message with Exit Information:**
  - The driver receives an entry message with a link for future exit payment.
  - Once this link is clicked, it will allow the driver to make the payment and allow 15 minutes for them to exit the gate and go thru the exit process.
1. **LED Display Update:**
  - The available parking space on the LED screen at the lot entrance decreases by one after the driver enters.

Check-out Process (Driver Exit)

1. **Payment**
  - The driver clicks the link provided in the entry message to make a payment before exiting.
1. **Approaching the Exit:**
  - As the car approaches the exit camera:
    - **If payment was made within the last 15 minutes** and the license plate is recognized, the gate opens automatically.
    - **If payment was made but the license plate is not recognized,** a speaker prompts the driver to show their QR code to open the gate.
1. **QR Code Fallback:**
  - If the camera cannot recognize the license plate or the gate fails to open automatically, the driver scans the QR code posted at the exit gate to proceed.
1. **License Plate Not Recognized (Fallback):**
  - If the license plate is not recognized and the camera or QR system fails, the driver can press the button under the QR code scanner to manually input their license plate number and open the gate.
1. **Additional Payment**
  - If the payment was made over 15 minutes ago or the driver hasn't paid, a speaker prompts the driver to complete the payment using the provided link or QR code.
1. **Gate Opens:**
  - Once the payment is confirmed, the system opens the gate for the driver to exit.
1. **LED Display Update:**
  - Upon exiting, the available parking space on the LED screen increases by one.

Error Handling (Driver Experience)

1. **Unrecognized License Plate**
  - If the camera fails to recognize the license plate during entry or exit, the system provides prompts via a speaker or directs the driver to manually input the license plate number using their phone or QR code.
2. **No Payment Detected:**
  - If no payment is detected when attempting to exit, the system directs the driver to complete the payment on the spot, ensuring the gate opens only after payment confirmation.

Developer Considerations

- **Real-Time License Plate Recognition:**
  - Ensure fast, accurate communication between the QR scan, camera trigger, and backend for smooth check-in and check-out processes.
- **Fallback for License Plate Failures:**
  - Implement manual input options and QR code scanning for license plate recognition issues, with clear driver instructions.
- **Grace Period Management:**
  - Provide a 15-minute grace period post-payment for the driver to exit the parking lot without additional charges.
- **Accurate Time and Rate Calculation:**
  - Maintain accurate time logs for each vehicle's entry and exit, ensuring proper billing according to the parking lot's rate structure.