Non-Functional Requirements (10)

(ie. Metric : mean time between failures. Measure : number of failures observed in a period of time.)

1. Product Requirements
    - 1.1. Security
        - 1.1.1. The system must assure that user data is secure. -- Access to driving or payment history must be prohibited *or at the very least limited to the user.*
            - 1.1.1.1. Metric : lifespan of a password
            - 1.1.1.2. Measure : number of logins will trigger a password reset
        - 1.1.2. NICE TO HAVE IF TIME PERMITS : Client-Server communication channels for devices must be secure.
            - 1.1.2.1. Data - Security of data transfer from user's device to our servers must remain tight.
            - 1.1.2.2. Payment - Connections between the user and the billing system must maintain a high-level of integrity and security.
    - 1.2. Reliability
        - 1.2.1. The Android-based system must perform its tasks (provide accurate parking data) as required and expected.
            - 1.2.1.1. Metric : defect rate based on the amount of hours the system is in operation.
            - 1.2.1.2. Measure : The system defect rate shall be less than 1 failure per 1000 hours of operations.
    - 1.3. Usability
        - 1.3.1. The system will be concerned about the ease of use and training end users.
            - 1.3.1.1. Metric : satisfaction ratio per type of user (ie. paid parking user vs non-paid)
            - 1.3.1.2. Measure : 80% < users polled after 3 months of usage shall give a satisfaction rating of 3 stars or greater on a scale of 1 to 5 stars
    - 1.4. Performance Efficiency
        - 1.4.1. The system must be able to handle concurrent requests from different users in proper transaction response times.
            - 1.4.1.1. Metric : number of transactions per unit of time
            - 1.4.1.2. Measure : The system must be able to process 100 payment transactions per seconds at peak times.
    - 1.5. Maintainability
        - 1.5.1. The system must be able to make updates quickly and cost effectively. Any new version must leave all database contents and personal settings unchanged.
            - 1.5.1.1. Metric : number of days of annually maintenance
            - 1.5.1.2. Measure : The system must not require more than 10 days of annual maintenance.
    - 1.6.
2. Process/Organizational Requirements
    - 2.1. Environmental/
        - 2.1.1. The system must not cause physical harm to users, non-users and surrounding fixed or moving agents (bystanders, buildings, city structures, personal or public property)
            - 2.1.1.1. Metric : rate of collisions as a result of using the system.
            - 2.1.1.2. Mesure : 99.9% of parking events registered with the system remain collision free.

3. External Requirements
    3.1. Legislative (see source : Gerald Kotonya & Ian Sommerville, Requirements Engineering, Diagram on page 3 of Sept 16 notes)
        3.1.1. The system must not infringe on road, off-road and parking regulations under the Ontario Highway Traffic Act and the City of Ottawa traffic bylaws
            3.1.1.1. Metric :
            3.1.1.2. Measure:
        3.1.2. Safety/security
            3.1.2.1. The app must not require user input while the user is driving.
                3.1.2.1.1. Metric: rate of occasions the system needed to be overridden due to emergency calls (ie calls to ambulance, police, fire department).
                3.1.2.1.2. Measure: 5% of the system's service may be accessed exceptionally to reach first responders should the user need emergency services from paramedics, police or fire services.

Functional Requirements (13)

4. For general/metadata view for any user
    4.1. Highlight new parking spots, the parking prices and/or rates.
    4.2. Display for the user the most updated information for parking prices.
    4.3. Display (on a map and/or a list) for the user all metered and non-metered parking areas registered with the app.
5. For app transaction info to the user
    5.1. Input user's vehicle licence plate number and reservation password (session ID) to the system's server for when users check in.
    5.2. Provide users with proof of payment when checking out of the app.
6. For user main interactions with the app.
    6.1. Create user profile with personal and vehicle details.
    6.2. Geolocate parking spot from list of options registered with the app.
    6.3. Display the details of the selected parking space (name, flat rate/price per minute, number of available spaces if applicable.)
    6.4. Reserve a parking spot in advance for a specific date, for a specific amount of time.
7. For managing the app's Backend
    7.1. Authenticate returning users before modifying any sensitive data.
    7.2. Accept reservation of parking space based on parking availability.
    7.3. Modify parking status : searching, parked, overstay, ticketed.
    7.4. Create a sessionID for each parking transaction and share it with the user.
    7.5. Provide users with a delay of time to complete parking reservation. Should the delay of time expire, the transaction is cancelled.

NOTES

7.6. Will need use of Google Maps (or other) for geolocating of all Ottawa addresses and streets.
    7.6.1. Nice to have - Turn by turn GPS directions to preferred/selected parking space. spo
    7.6.2. Route calculation.
    7.6.3. Route optimization.