M2

1. Use Cases

    I.

    Use case: Approve or Decline a lift request
    Primary Actor: Admin
    Scope: accept or decline a lift request
    Brief: admin approves or decline lift request made by user
    Stakeholder: admin, user
    Post conditions: database updated, user gets notify
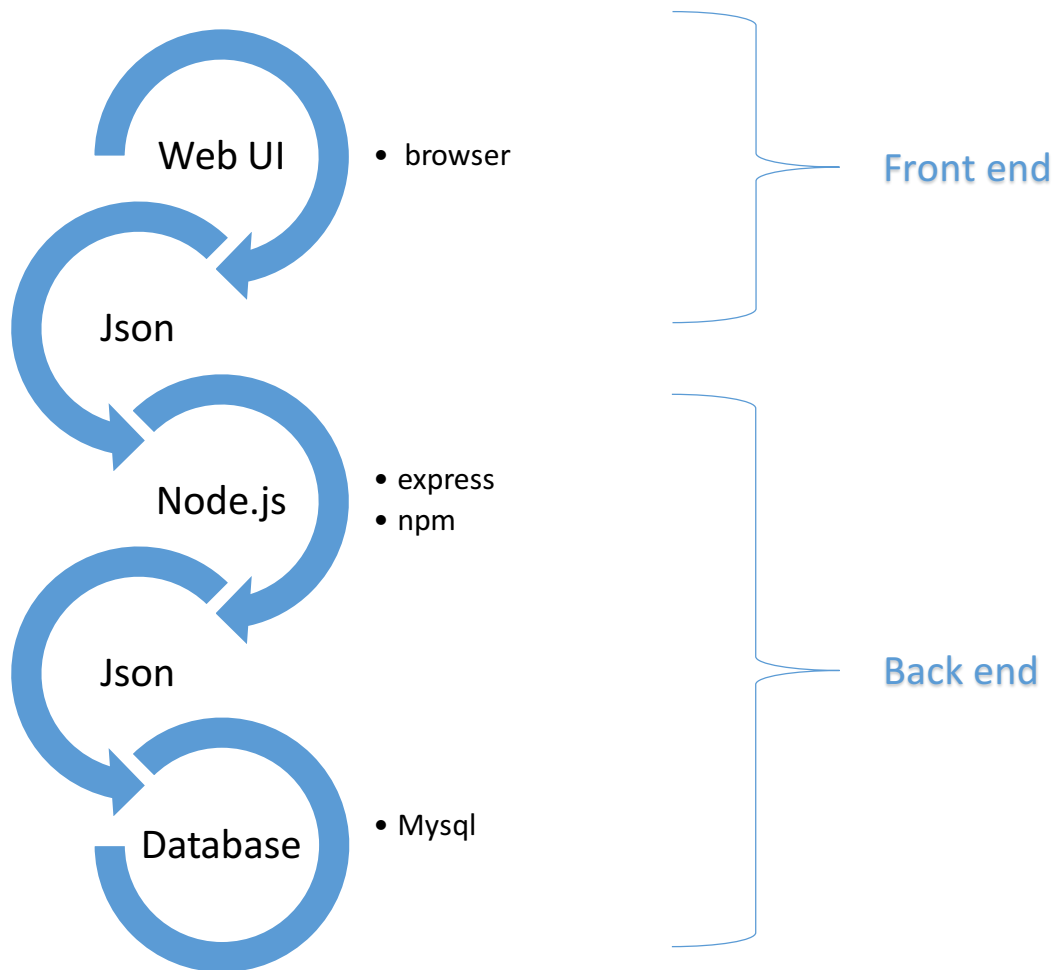   **Preconditions**: admin account, request form submission
    Triggers: user submit a request
    Basic Flow:
        a. Admin gets notify after every request made
        b. Admin accept or decline the request
        c. User will be notified through email about the status of their request.

2. Software Architecture

Web UI • browser
Json
Node.js • express • npm
Json
Database • Mysql

Front end

Back end

3. API
   Lyft Request Submission
   {
   "action": " submit a request"
   "name": "Fousseini konate"
   "email":" fkonat@pointpark.edu"
   "department": "athletic"
   "date":"02/11/18"
   "time":" 02:00 PM"
   "destination": "CCAC"
   "passager_number":"10"
   "return_time":" 06:00PM"

```
}
=>
{
        "Success": "true"
        "message": "Form submit with success!"
}
or
{
        "Success": "false"
        "message": "there was a problem"
}
```

4. Security
   a. We will Enable https by configuring nginx.
   b. Personal information (in our case we will focus on the passwords, budget id)
   c. We will hash them
   d. We will use (Nginx, certbot, some npm such as crypto)
   e. Prevent database injection from inputs