## Project Description

E-Parc is an app that improves campus parking for staff, students, and visitors by providing real-time parking space information. Users of the app can discover available parking spaces and navigate to the corresponding spaces using GPS technology. E-Parc is designed with simplicity, effectiveness, and reliability, especially during rush hours when it takes lengthy and frustrating minutes to locate an open parking space. The system aims to reduce congestion, save time, and improve campus mobility through user-friendly design and innovative technology.

## Requirements Summary

| | **Processor Cores** | **OS** | **RAM** | **Storage** |
|---|---|---|---|---|
| **MINIMUM REQUIREMENTS** | **Dual core** | **Android 5.0 (Lollipop)** | **2 GB** | **300 – 400MB free space** |
| **RECOMMENDED REQUIREMENTS** | **Processor Cores** | **OS** | **RAM** | |
| | **Quad core, Octa core** | **Android 9.0 (Pie)** | **4GB and higher** | **600-700 MB free space** |
| **OTHER REQUIREMENTS** | **Permissions** | **Camera(for QR scan), Location, storage, notifications** | | |

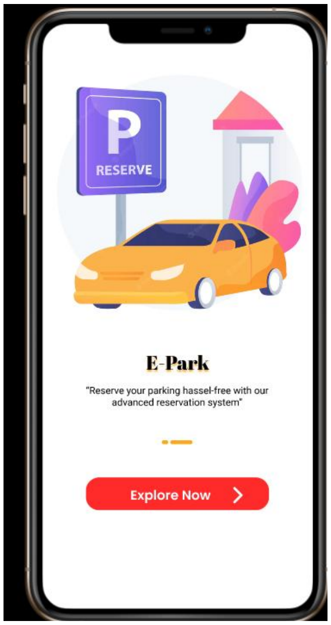
## Prototype Description

E-Parc is a mobile application that we built to assist students, faculty, and visitors in quickly locating a real-time available parking spot on campus. E-Parc came about from the idea of constantly circling to locate a parking spot, particularly during busy times. Through user-centered design principles, we ensured that the app functions on various devices and is simple to use. The purpose of E-Parc is to make parking less painful and more efficient for the user by providing updated, accurate, and reliable parking information, so users do not waste time or rely on signs that are more than likely outdated.

### Overview of the Prototype

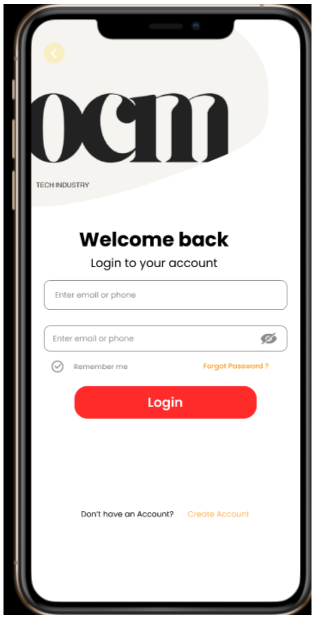
The prototype for E-Parc was created using Figma and simulates a fully functional mobile app. It includes the key features identified in the requirements phase such as real-time parking updates, a zone map, GPS integration, and a minimal-tap user interface.

### Detailed Screens with Description



E-Park

"Reserve your parking hassel-free with our advanced reservation system"



Welcome back

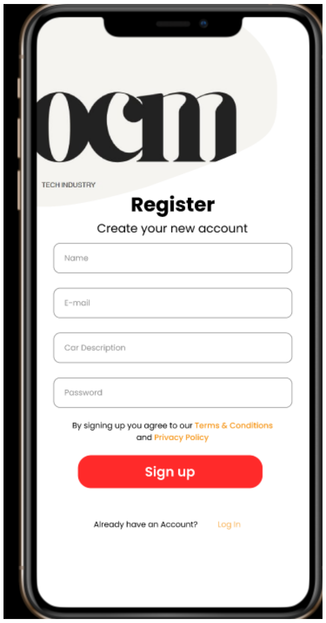
Login to your account

Enter email or phone

Enter email or phone

Remember me Forgot Password ?

Don't have an Account? Create Account



Register

Create your new account

Name

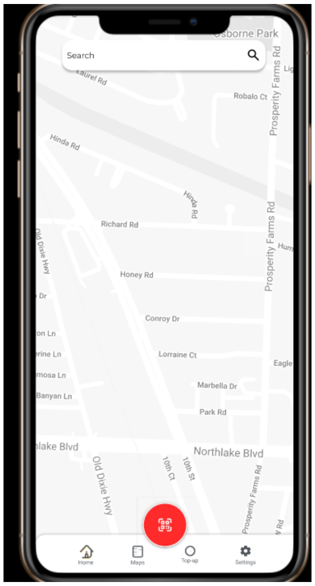
E-mail

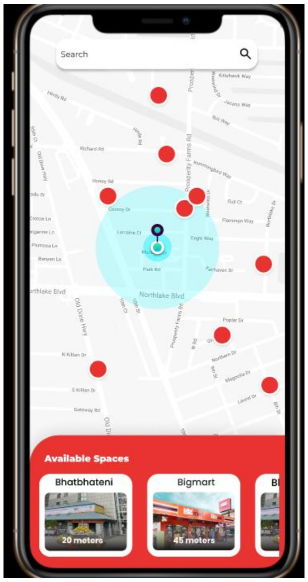
Car Description

Password

By signing up you agree to our Terms & Conditions and Privacy Policy

Already have an Account? Log In

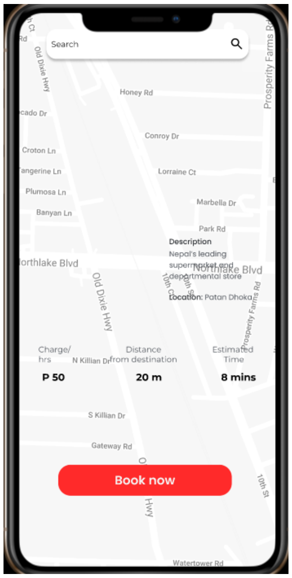




Available Spaces

Bhatbhateni
20 meters

Bigmart
45 meters



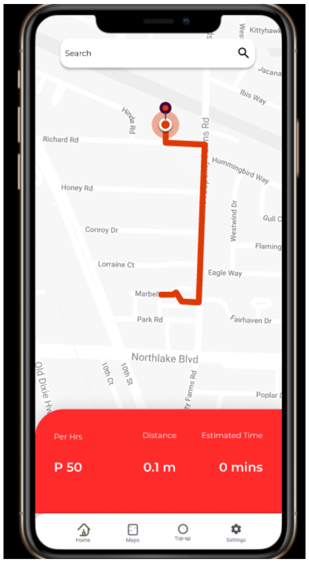
Description
Nepal's leading supermarket and departmental store
Location: Patan Dhoka

Charge/hrs
P 50

Distance from destination
20 m

Estimated Time
8 mins

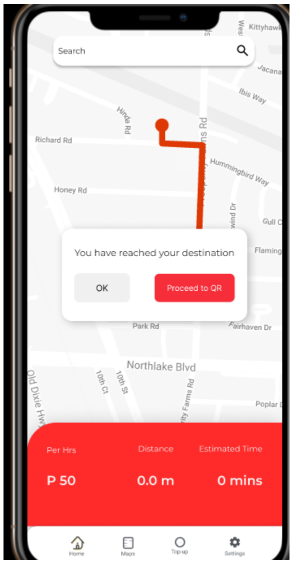
Book now



Per Hrs
P 50

Distance
0.1 m

Estimated Time
0 mins



You have reached your destination

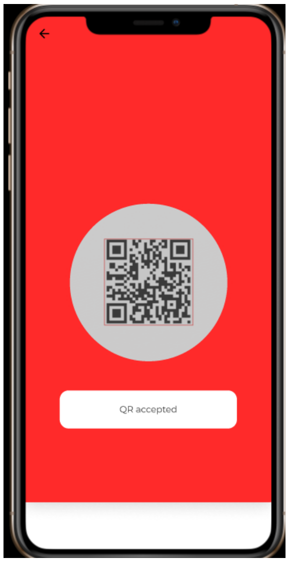
OK

Proceed to QR

Per Hrs
P 50

Distance
0.0 m

Estimated Time
0 mins

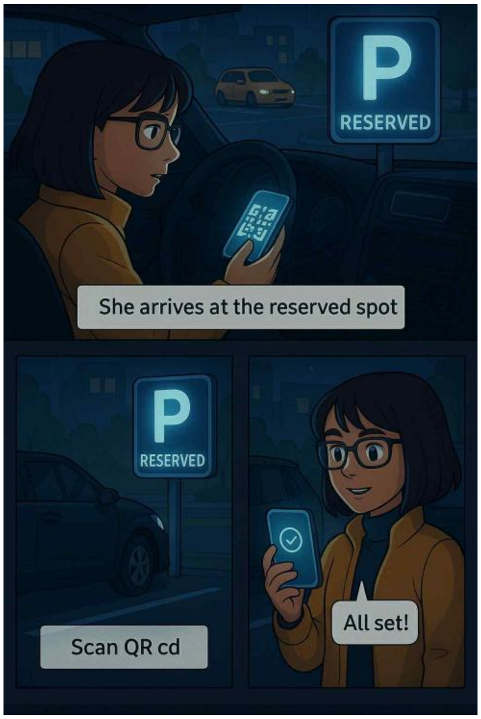


QR accepted

https://www.figma.com/design/hYdSlGJgslOKaZZRSE5ACZ/OpenPark--Community-?node-id=2-2&p=f

## SCENARIO

There has to be a better way..
P
Book
Mia remembers downloading the SmartParking app
$5
Book
Nice! That one's just a block away
That was seamless!
That was seam-less!



**1. Home Screen**

Displays a summary of all parking zones with color-coded availability indicators:

- Green: Available
- Yellow: Limited spots
- Red: Full
  *Tapping a zone leads to the zone detail screen.*

**2. Zone Detail Screen**

Shows the zone layout on a mini-map with available spots marked. Also contains:

- Estimated walk time to key buildings
- Last update time
- Button for “Navigate”

**3. Navigation Screen**

Uses embedded Google Maps to show directions from the user’s current location to the selected parking zone.

**4. Parking History (Optional)**

Shows previously used parking zones, date/time, and whether the session was successful.

**User Scenario**

Maria, a nursing student running late for her 9 AM class, opens the E-Parc app. She sees that Lot A is full (red), Lot B is nearly full (yellow), and Lot C is available (green). She taps Lot C, clicks “Navigate,” and follows GPS directions. She arrives on time and avoids the usual 10-minute delay of circling the campus.

## Rationale for Prototype Selection (20 pts)

We chose this prototype because it prioritizes user efficiency and aligns with core user needs like speed, clarity, and low cognitive load. Its **advantages** include:

- Minimal interaction steps
- Visually intuitive status indicators
- Compatibility with real-time data updates
- Easy integration with existing navigation tools

**Disadvantages**:

- Dependent on accurate real-time data feeds
- Limited offline functionality without caching setup
- Simulated prototype lacks back-end integration at this stage

Despite these limitations, the design meets the core usability and functionality needs required for evaluation and testing.

## Changes to Requirements During Prototyping (20 pts)

As we developed the prototype, several requirements evolved:

- **Added Offline Mode (New)**: Based on user feedback, we added offline caching to help users in areas with weak signals.
- **Modified Navigation Flow**: Originally, navigation launched in a separate app, but was changed to open within the app using an embedded map view for smoother experience.
- **Dropped In-App Chat (Removed)**: An early idea included a “community help” chat feature, but it was removed due to complexity and distraction from the core task.

These changes came from iterative user feedback and internal design testing using rapid prototyping.

**Initial Evaluation Plan**

We plan to evaluate the E-Parc prototype using three key usability techniques:

1. **Heuristic Evaluation**
   A small group of usability experts will evaluate the prototype based on Nielsen's 10 usability heuristics (e.g., visibility, match with real-world concepts, error prevention).
2. **Cognitive Walkthrough**
   We will simulate the experience of new users completing tasks (e.g., finding a spot and navigating to it) to identify where confusion or inefficiency occurs.
3. **User Testing (Scenario-Based)**
   We will ask real users (students/faculty) to perform tasks on the prototype while noting:
   - Time taken to find a spot
   - Number of taps per task
   - User satisfaction scores post-task

**Criteria to Measure Effectiveness:**

- Task success rate
- Average completion time
- User error frequency
- Perceived ease of use (via post-task survey)

These methods will help validate the system's usability and inform improvements for a future working version of E-Parc.