SIM UOW ISIT307

ASSIGNMENT 2

Last updated: 13 Feb 2024

Timothy Mah

txamah001@mymail.sim.edu.sg

8750634

Tutorial Group T01F

Contents

[Introduction 2](#_Toc158732423)

[Usage 2](#_Toc158732424)

[Structure 2](#_Toc158732425)

[File structure 2](#_Toc158732426)

[Flow 2](#_Toc158732427)

[Screenshots 5](#_Toc158732428)

[Admin type user 5](#_Toc158732429)

[Normal user 8](#_Toc158732430)

[Database 11](#_Toc158732431)

# Introduction

Hello this is my documentation for SIM UOW ISIT307 ASSIGNMENT 2.

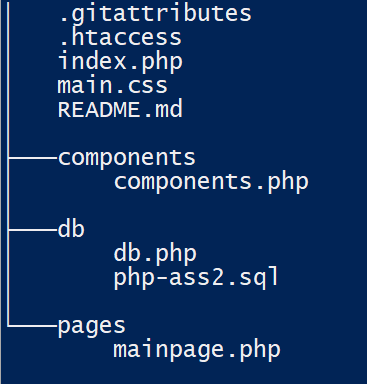
# Usage

Set up database

Access localhost in browser

# Structure

## File structure



* .htacess
  + This file allows me to configure the “starting point” of the website to be index.php
* Index.php
  + The “starting point” of the website. It’s the controller component of the MVC design that my website follows
* Main.css
  + All css code is stored in one file
* Components
  + Components.php
    - All functions that do not do processing but return html code
* Db
  + Db.php
    - Database connection is handled here
  + Php-ass2.sql
    - .sql file to create the database needed for my website
* Pages
  + Mainpage.php
    - The ‘main’ html page. Index.php will send components and all relevant data to mainpage.php

## Flow

The system design of my website is the same as in assignment 1. When the user enters the website, index.php will be the first file “called”. Then index.php will serve the components and relevant data for each route

All functions with the term “Component” in them only return HTML code. They do not perform any processing or logic. They are called multiple times in each route however I will omit their explanation.

For all search inputs, user can enter partial search term. If user enters a blank search input, all results will be shown.

Routes (in order of what’s written in my code)

/

This is the main route. User must be logged in to access. Else redirect to /login

If user is admin, the admin page features are displayed. Else if user is normal, normal user features are displayed

/users

This is the first feature of admin type account. User can view all user accounts and search for user account. Search can be partial. Search by name or ID.

There are 2 functions used. displayAllUsers() and searchForUser($searchUserName). These are similar functions but what’s different is the SQL statement used. Perhaps I could have merged the two functions. As their name suggests, displayAllUsers() uses a SELECT \* statement however searchForUser($searchUserName) uses a SELECT \* WHERE type statement for searching.

/ploc

Short for parking location. Very similar to the /users route however from this page, user can also add parking location.

Like in /users, there are 2 functions for searching and displaying parking locations. displayAllPlocs() and searchploc($searchploc).

The /insertparkinglocation route is “called” to add new parking location.

/userploc

This is the page where normal type user can search for available parking location. There are 2 functions used. displayPlocsNotFull() and plocnotfullsearch($searchLocation). They are similar functions that differ by SQL statements. The former uses a SELECT \* WHERE capacity > 0 to show all available parking location while the latter uses SELECT \* WHERE capacity > 0 AND xx LIKE $ searchLocation.

/plocfull

This is the page where admin type user can view and search for parking locations that are full. 2 functions are used. Plocfullsearch() and displayPlocsFull(). They are both similar functions but differ in SQL statement. The searching function uses more WHERE xx LIKE conditions to perform search

/plocnotfull

This is the page where admin type user can view and search for parking locations that are not full. 2 functions are used. plocnotfullsearch($searchLocation) and displayPlocsNotFull (). They are both similar functions but differ in SQL statement. The searching function uses more WHERE xx LIKE conditions to perform search

/insertparkingloc

This is the page that processes adding new parking location from the insert parking location form in /ploc. Once it has done processing, it redirects back to /ploc

/check

This is the page where admin type user can view and search for checked-in parking locations. 2 functions are used. printCheckIn() and searchCheckedInLocation($searchName). They are both similar functions but differ in SQL statement. The searching function uses more WHERE xx LIKE conditions to perform search

User can also check in and check out from this page. When checking out, the /checkout route is “called”.

/usercheck

This is the page where normal type user can check in at an available parking location. When user checks in, the /checkin route is “called”. After checking in, the expected checkout time, number of hours to checkout, cost per hour, and Cost per hour for late checkout, are all displayed.

/history

This is the page where normal type user can view all previously checked out parkings. The searchCheckedInLocationByName($currentUserName) function is called to display the table.

/current

This is the page where normal type user can view all checked-in parkings that have not been checked-out. The activeparkingsbyname($currentUserName) function is used to list the table. The user can also checkout the corresponding parking from this table. The /usercheckout route is “called” to check-out.

/checkin

This is the route that processes check in. It does not return a webpage.

/payment

This is the page where normal type user can view the “payment” amount of their current checkout. If user has not checked-out during their current sitting, then the page is blank.

/checkout

This is the route that processes check out. It does not return a webpage.

/usercheckout

This is the route that process check out but for user type account. It does not return a webpage.

/css

This route returns the css file

/register

This route returns the webpage for anyone to register a new user

/registernew

This is the route that processes new user registration. It does not return a webpage

/login

This route returns the login web page

/loginuser

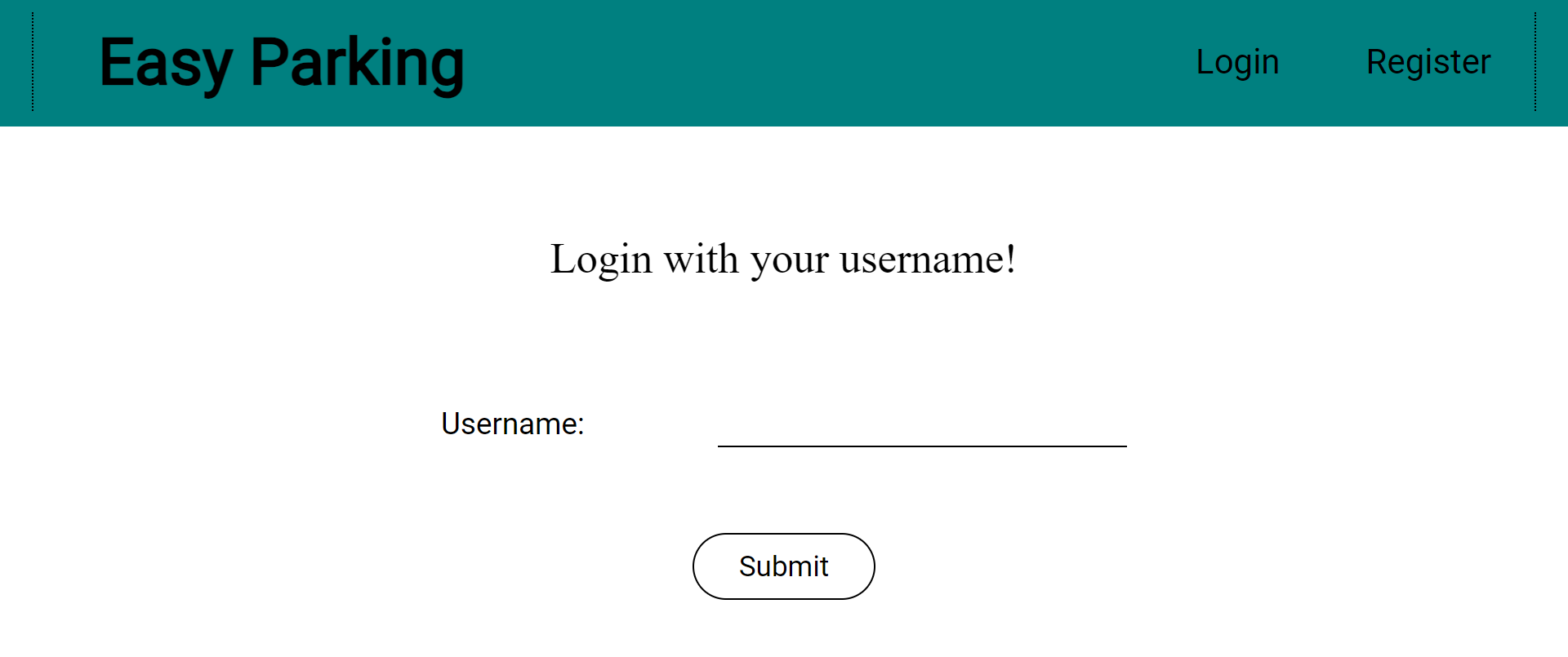
This route processes the login. It does not return a webpage

/logout

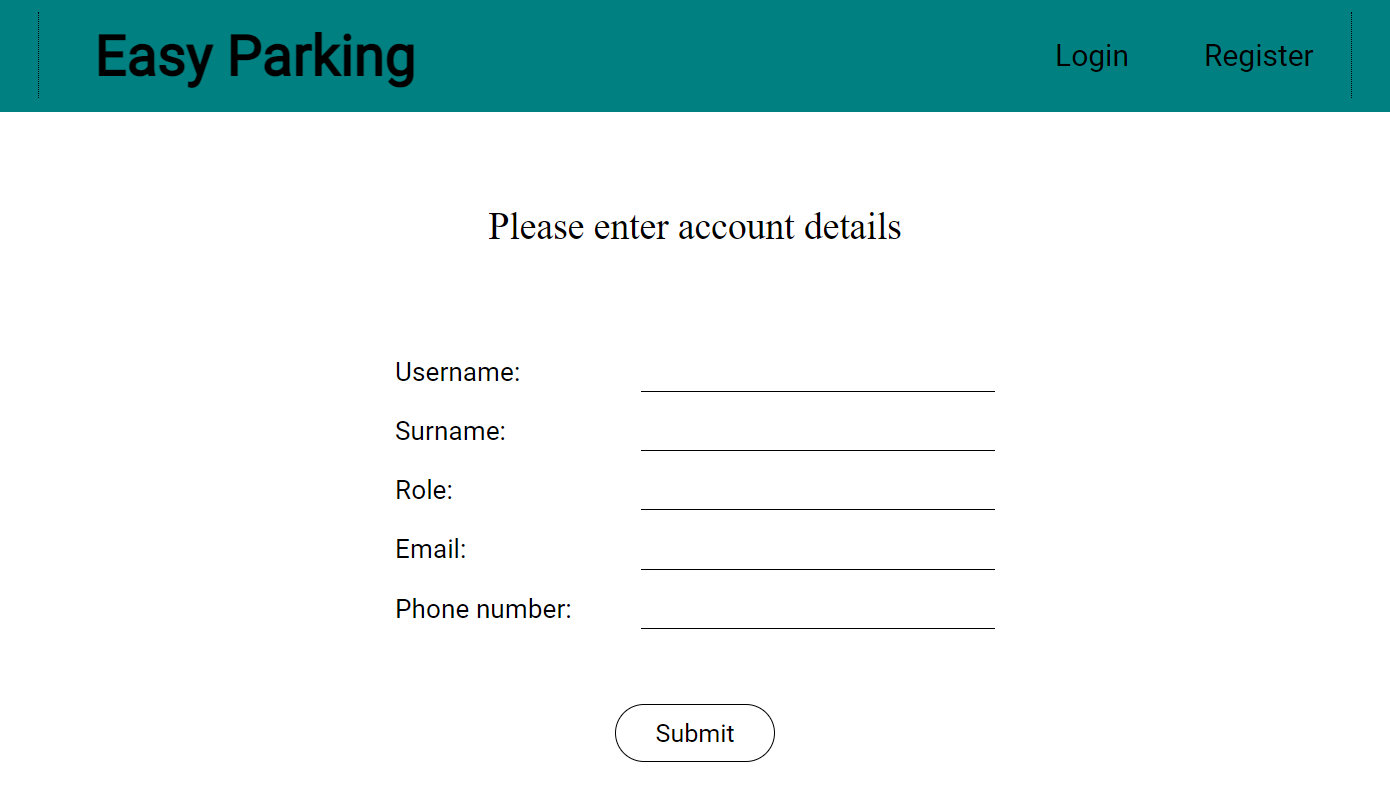
This route processes the log out. It does not return a webpage. Destroys all relevant sessions.

## Screenshots

This is the first page that any user would see

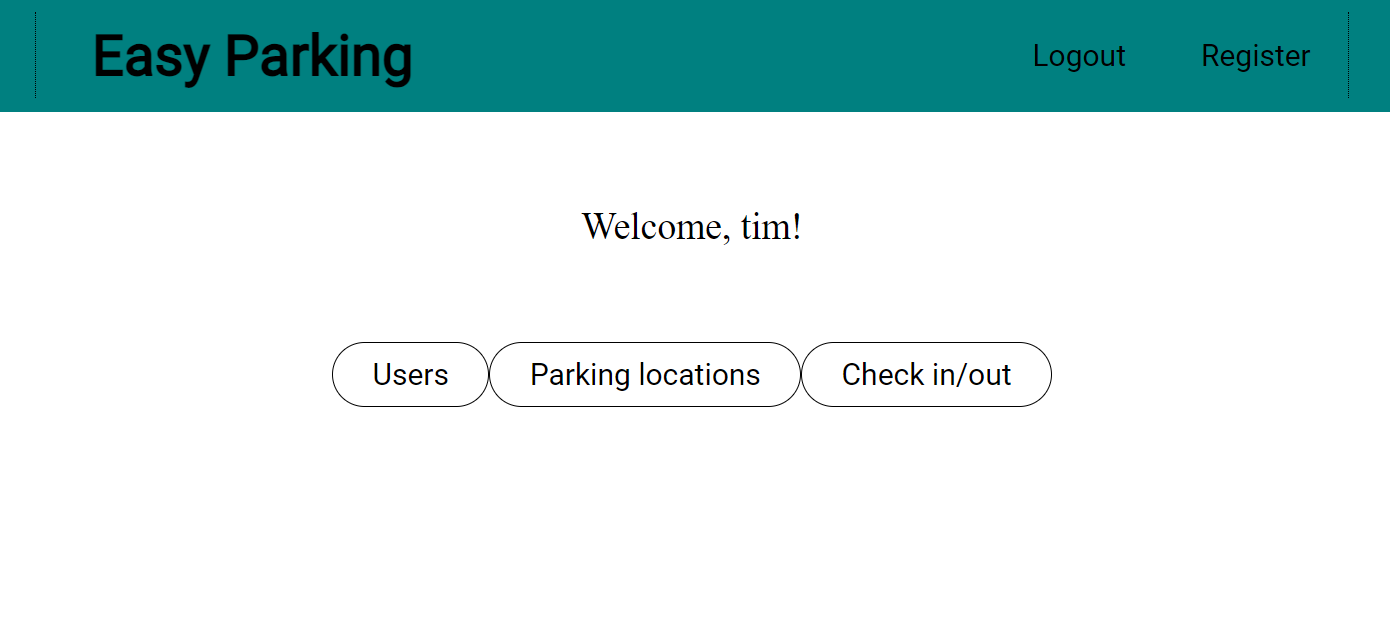


The user may register a new account after clicking the Register button in the header

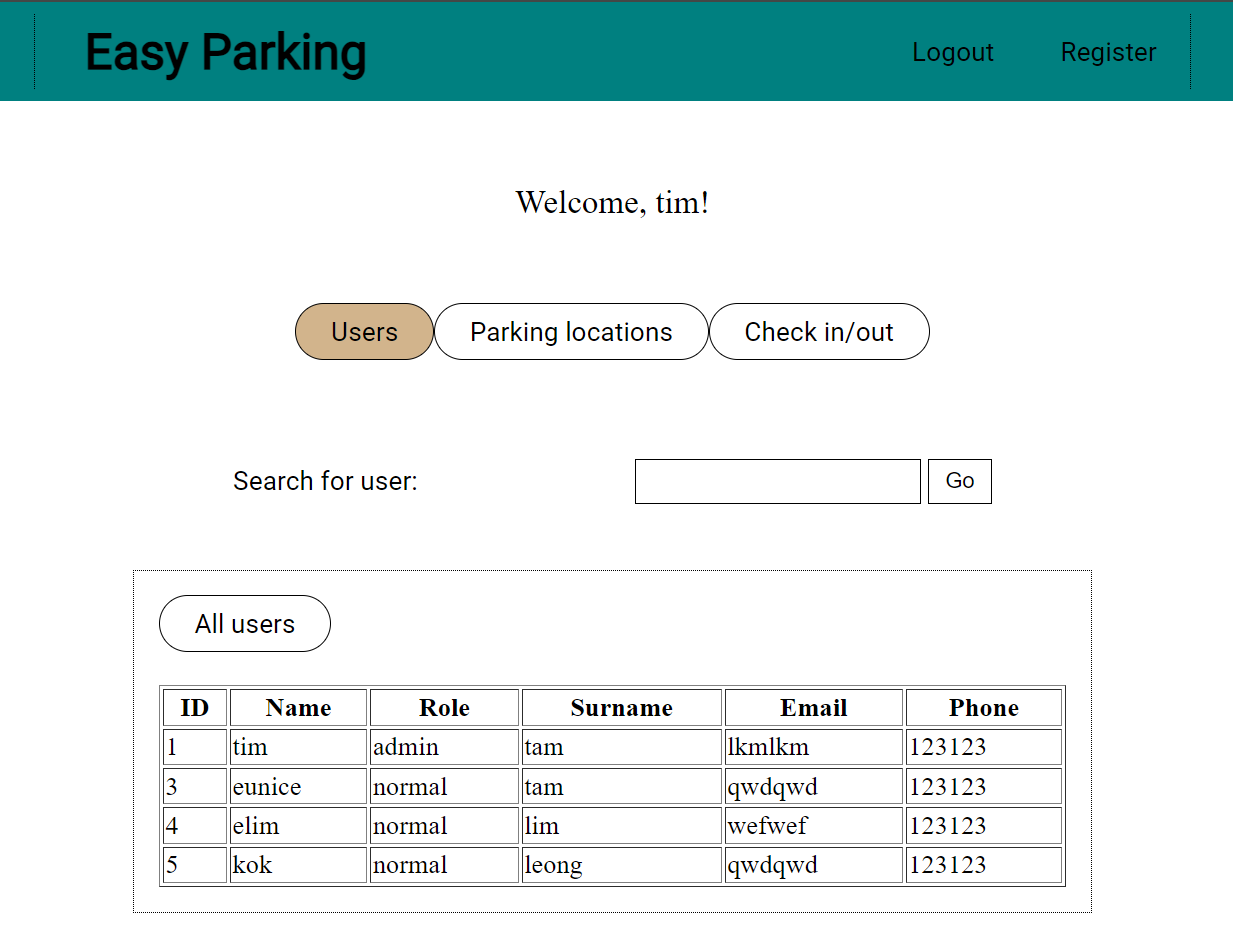


### Admin type user

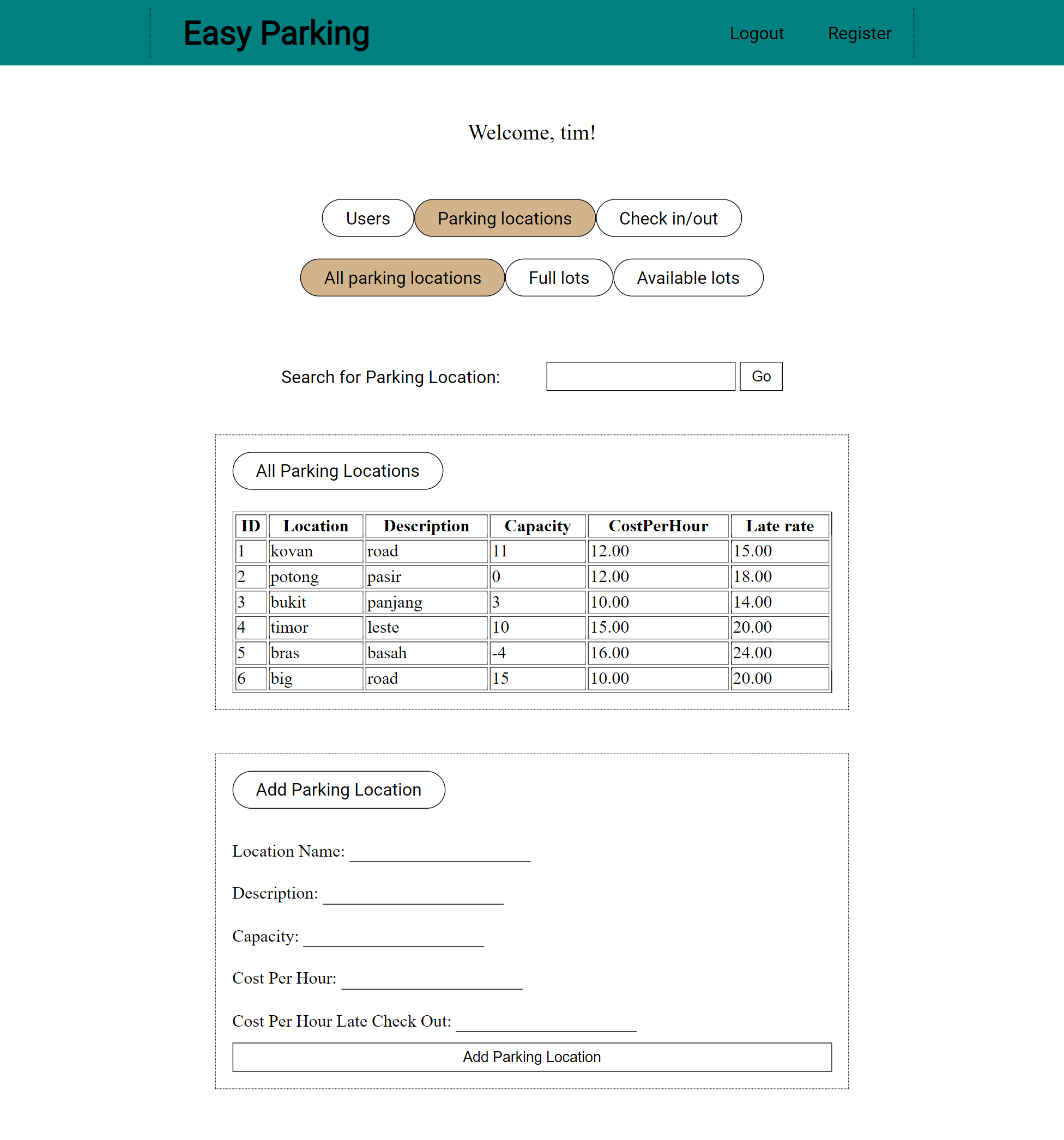
This is the page an admin type user would see after logging in



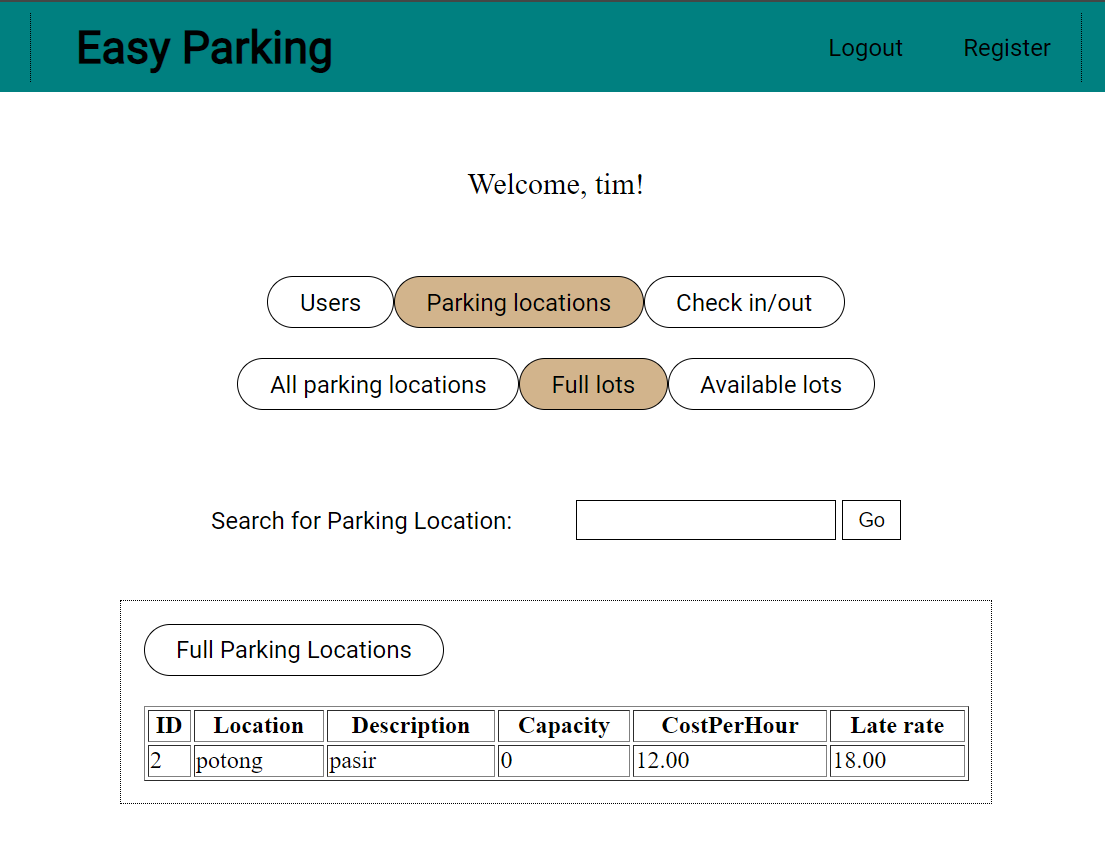
When user clicks on Users, the admin can view and search for all users



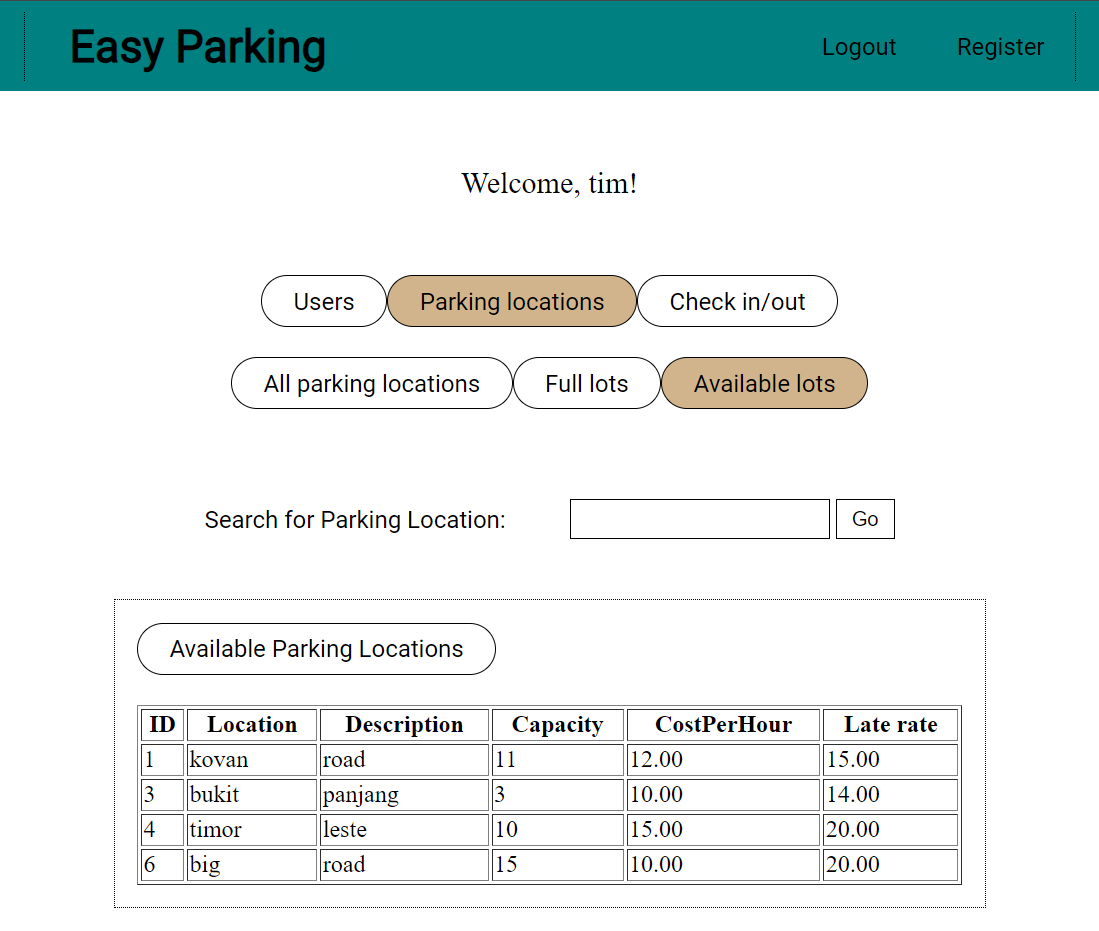
When user clicks on parking locations, user can view and search for all parking locations. User can also add parking location



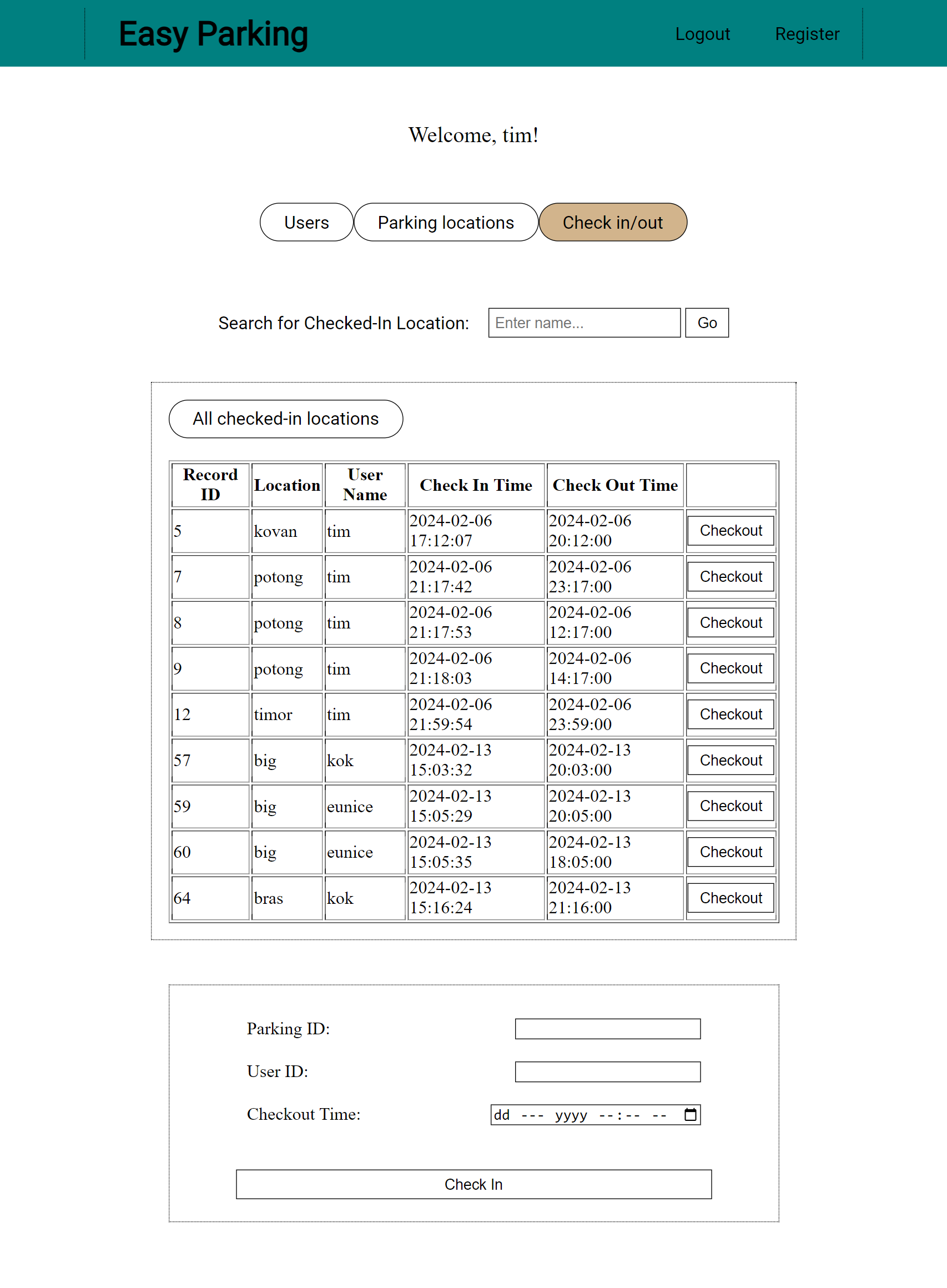
User can view full parking locations. Can view all or search.



User can view and search for available parking locations

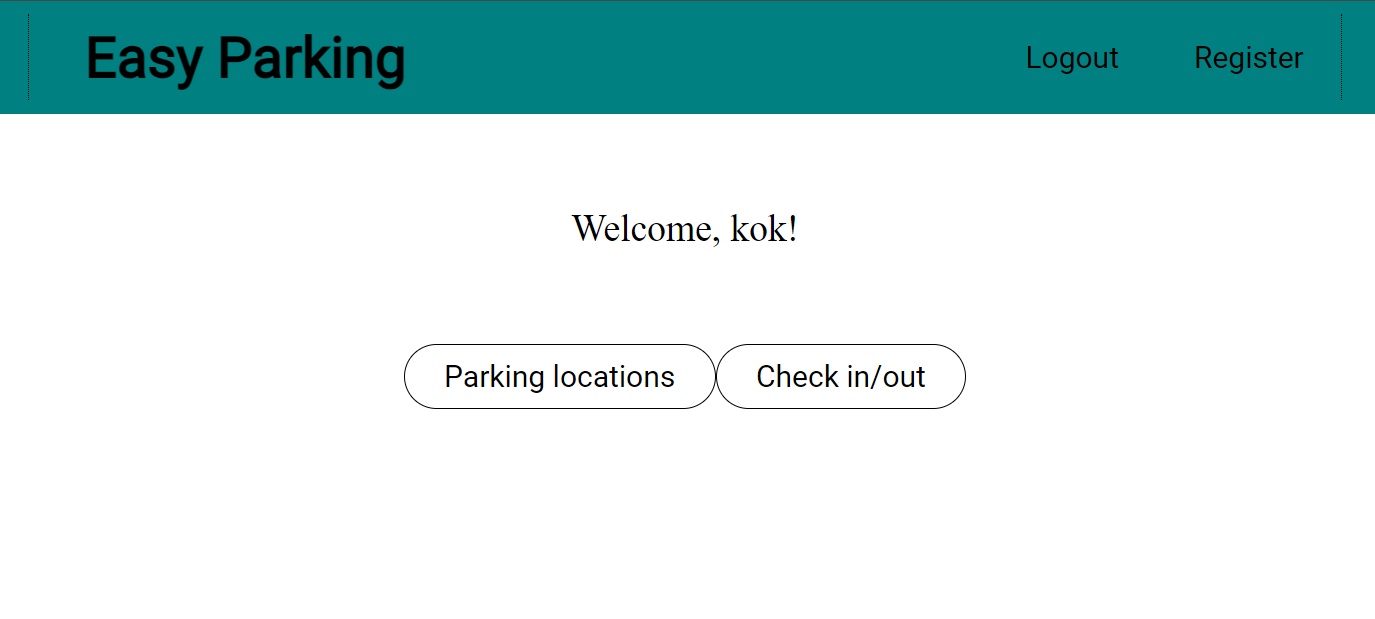


When user clicks on check in/out, user can view or search checked-in parking locations. Can checkout parking location. Can check in parking location.

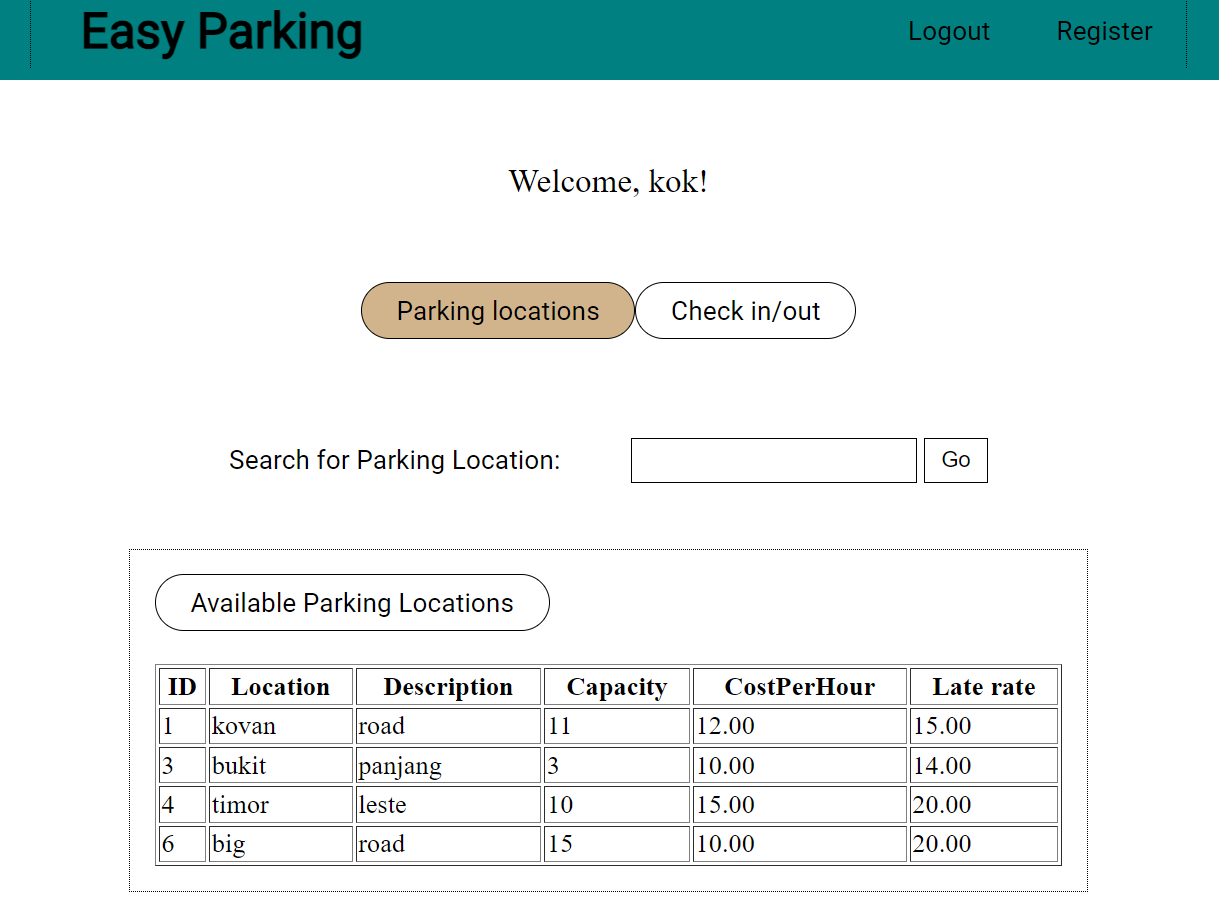


### Normal user

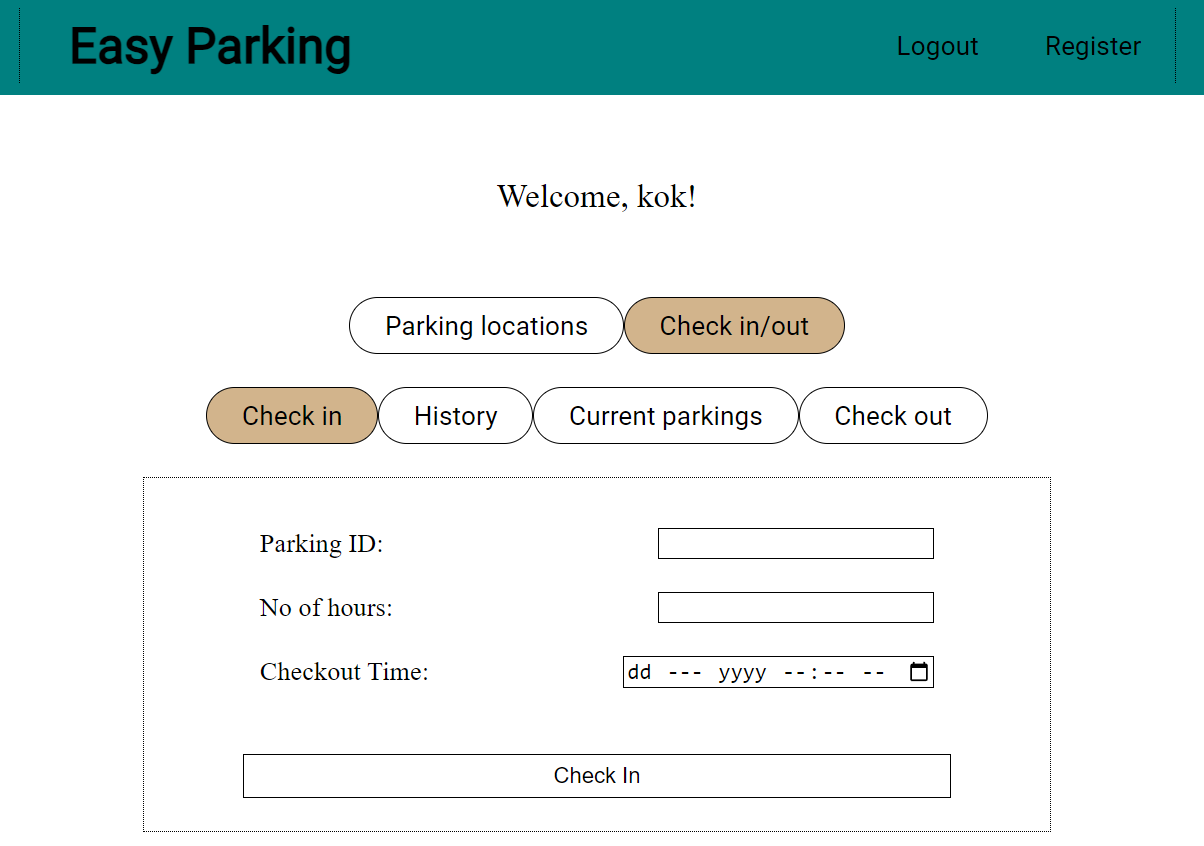
This is the first page a normal type user would see after logging-in



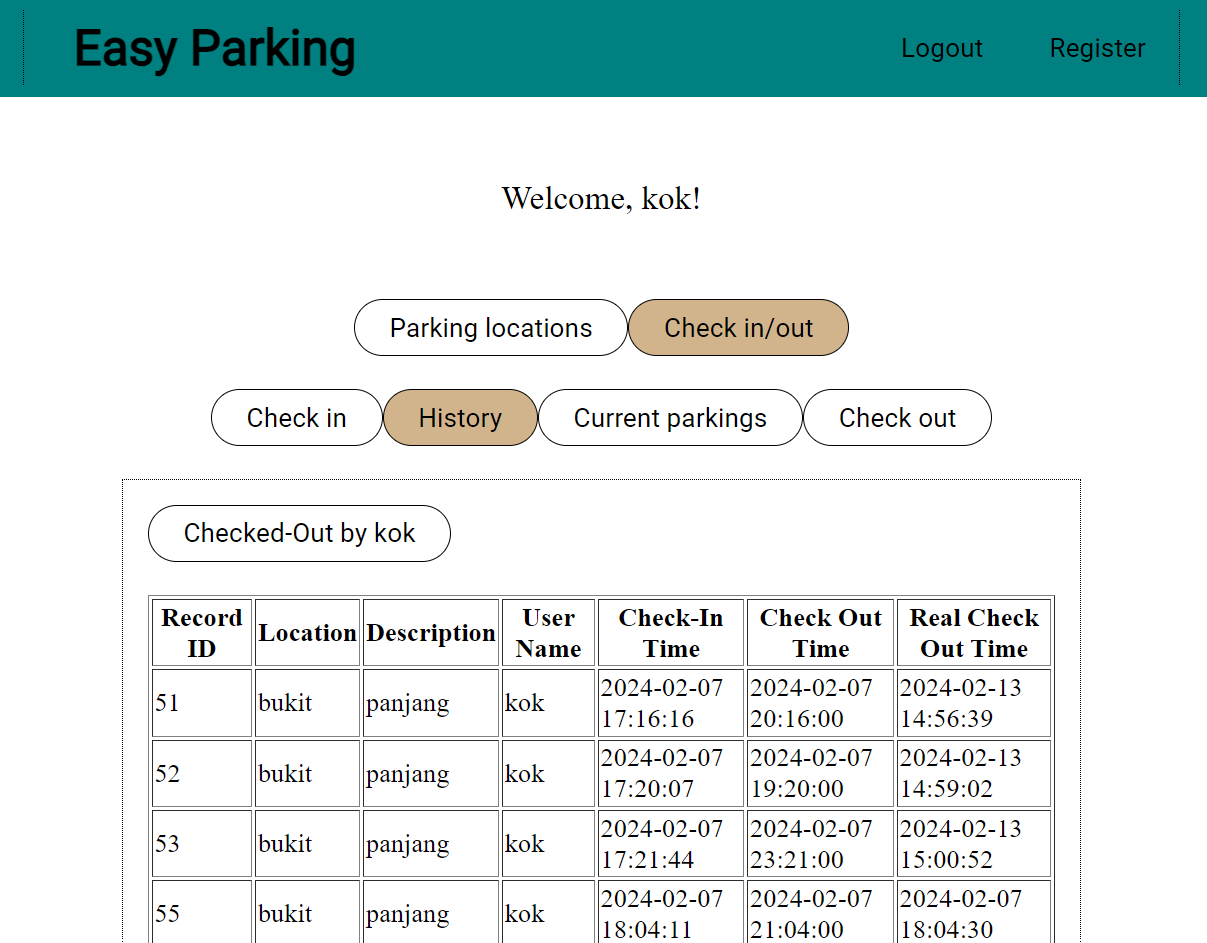
When user clicks on parking location, user can view or search all available parking locations.



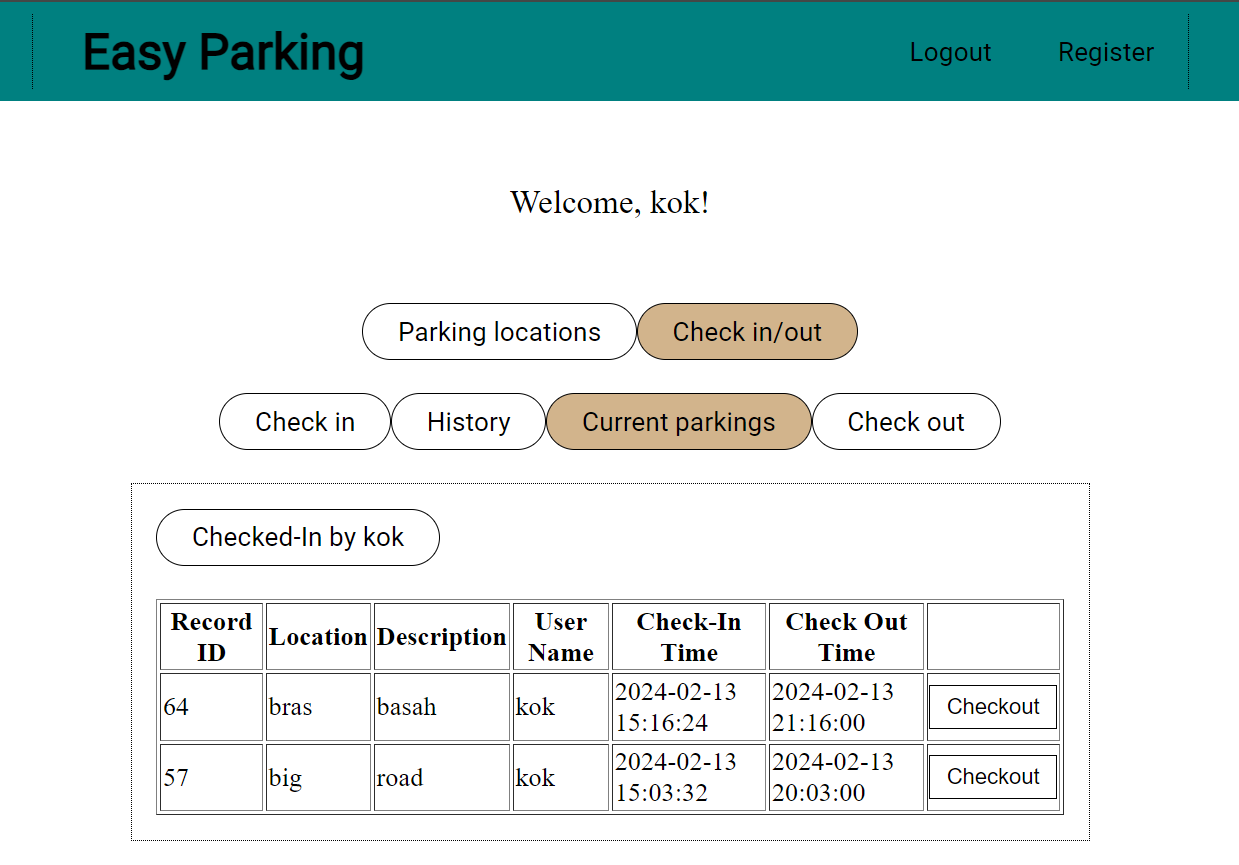
When user clicks on check in/out, user is sent to the check in tab. From there, user can check in to an available parking location.



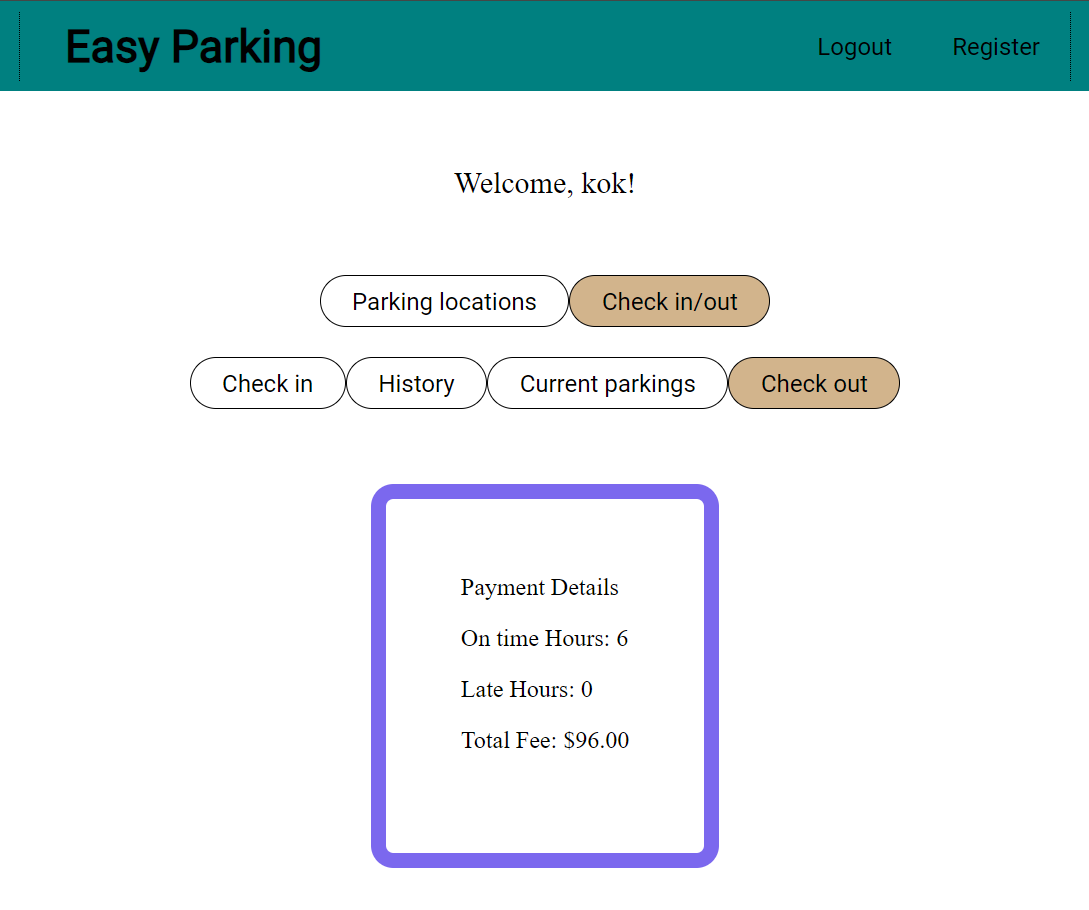
When user clicks on history, user can view all previously checked out parkings



When user clicks on current parkings, user can view all parkings checked-in but not checked-out



When user clicks on check out, it is blank unless user has already checked-out a parking at the current sitting



## Database

There are 3 tables used. Usertable, parkingrecords, and parkinglocs. Each table designed as per assignment requirement.

