



HELLO VELO

Kim Alizadeh, Bahar
Jabarzadnouri, Tyler Smalley, Aly
Westwood, Kate Post
April 8, 2022

BUSINESS OVERVIEW

BIKE RENTAL SERVICE

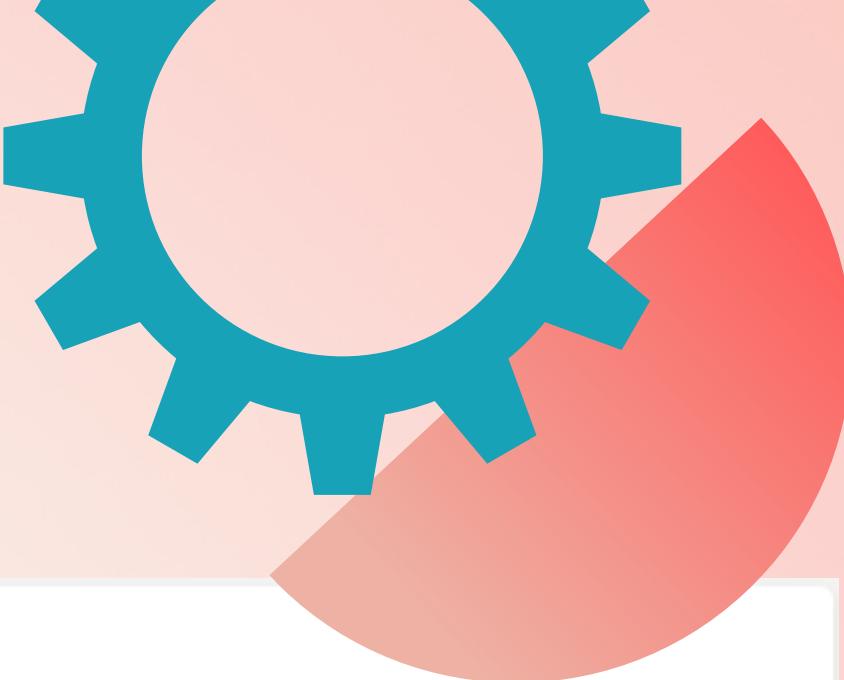
Bike tracking

Subscription and payment

Promotions

Administrator functionality





SUBSCRIBING AND DISCOUNTS

SOLUTION OVERVIEW

 **Register Payment Info**

Card Info

 4242 04 / 24 242 42424

Promo Code

NWO5GJ09

REGISTER CARD

Subscription/Billing

Start Date: **2022-04-07**
Type: **BIKE**

This invoice will be finalized on the due date.
Subscription Price: **\$18.00 per month**
Overage Price: **3\$ per 30 minutes**
Overage Minutes: **416**
Overage Fees: **\$42.00**
Rental Station Discounts: **-\$1.00**
Applied Promos:

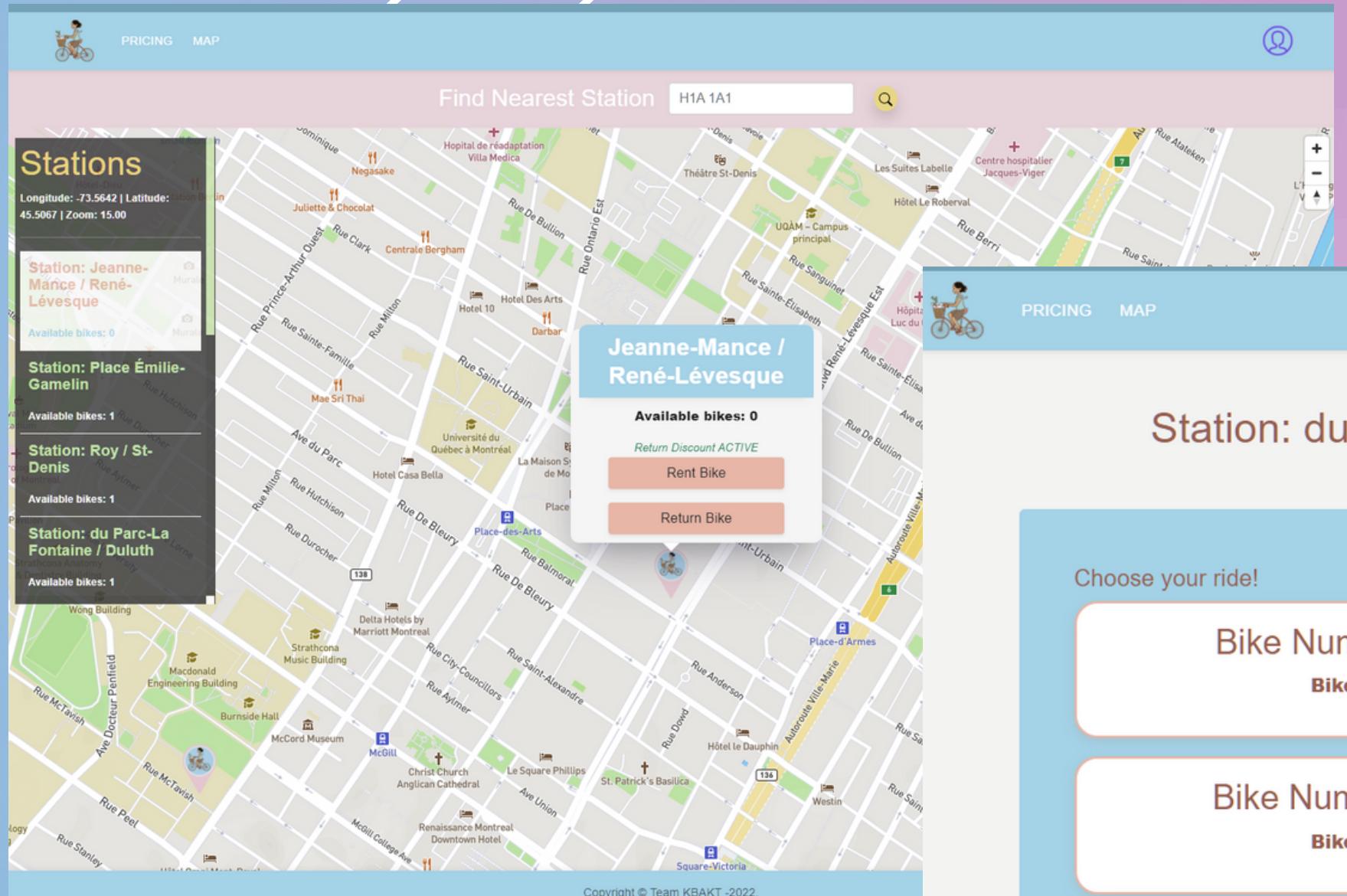
- **Referral Discount - \$5.00 off**
- **Referral Discount - \$5.00 off**

Current estimate for next invoice: **\$49.00**
Next Payment Date: **2022-05-07**
Previous Invoices:

- [Download Invoice 1](#)

Cancel Subscription

Station Map & Rentals



Find Nearest Station H1A 1A1

PRICING MAP

Jeanne-Mance / René-Lévesque

Available bikes: 0

Return Discount ACTIVE

Rent Bike

Return Bike

Choose your ride!

Bike Number: LL10202021

Bike Type: ELECTRIC

Bike Number: AA12345677

Bike Type: ELECTRIC

Bike Number: BB12345678

Bike Type: ELECTRIC

RENT BIKE

Copyright © Team KBAKT -2022.

Station: du Havre / de Rouen

PRICING MAP

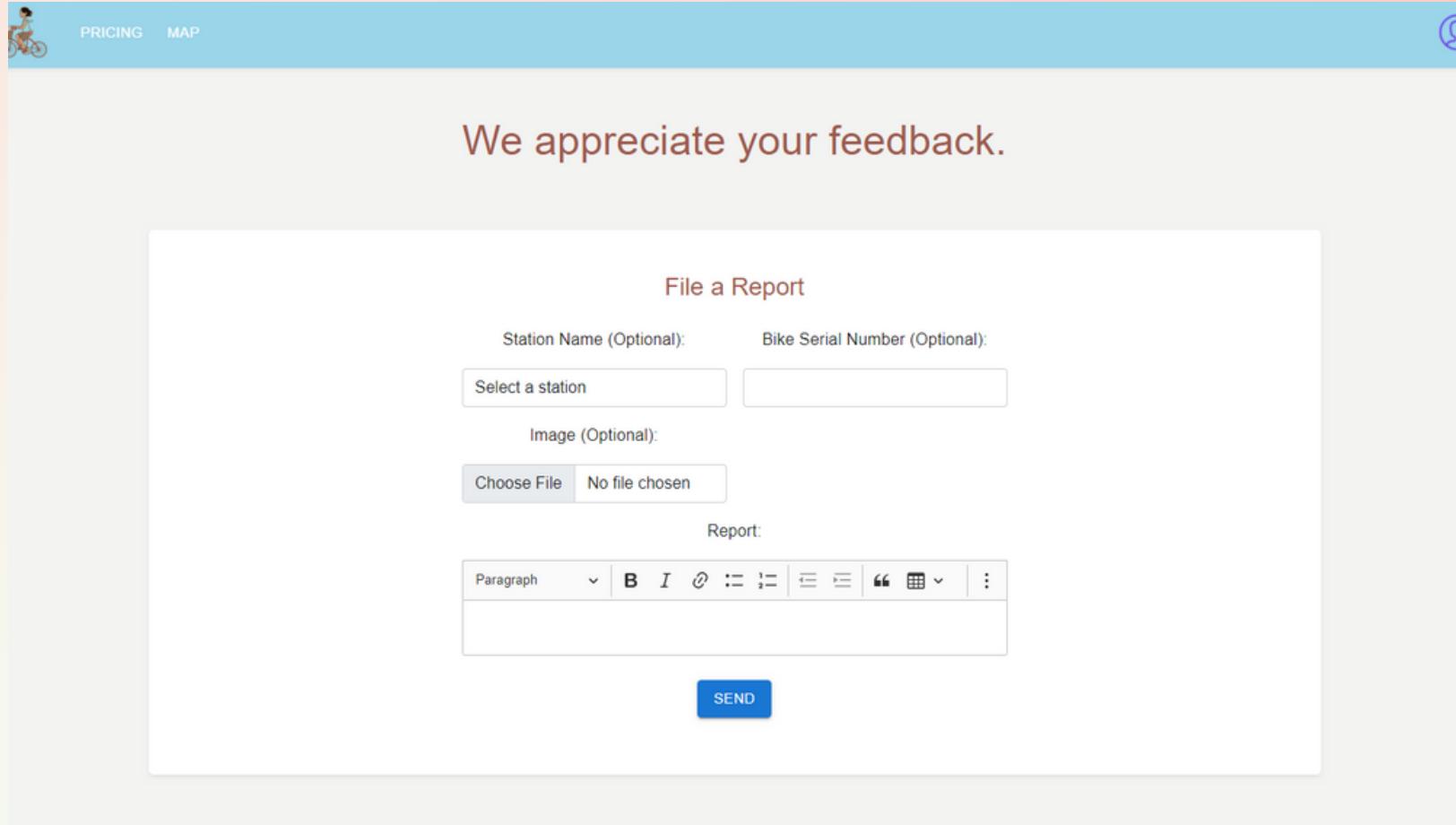
Your Current Rental:

Bike Serial Number
BB12345678

Start Time:
5:23:29 AM, Thursday April 7th 2022

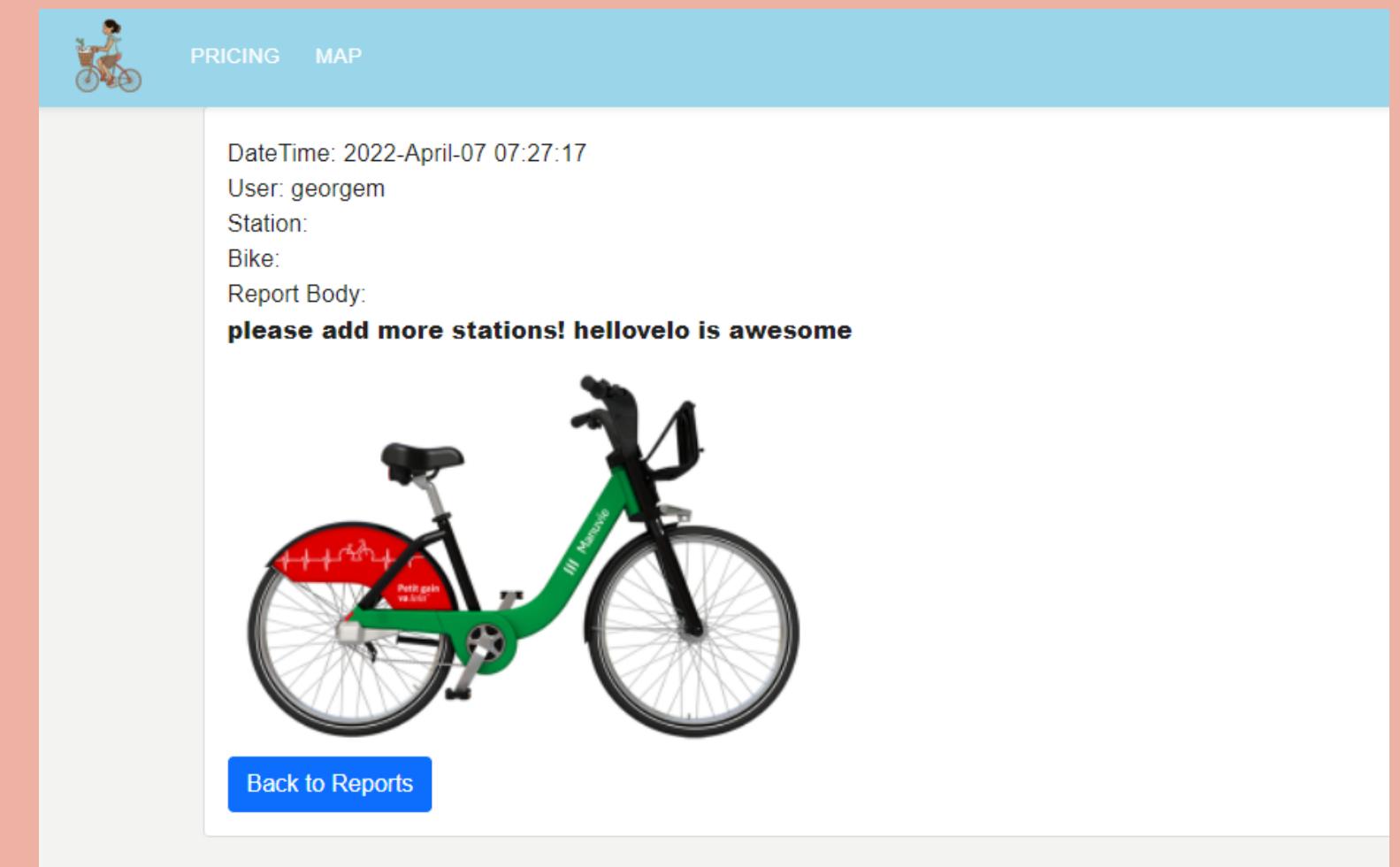
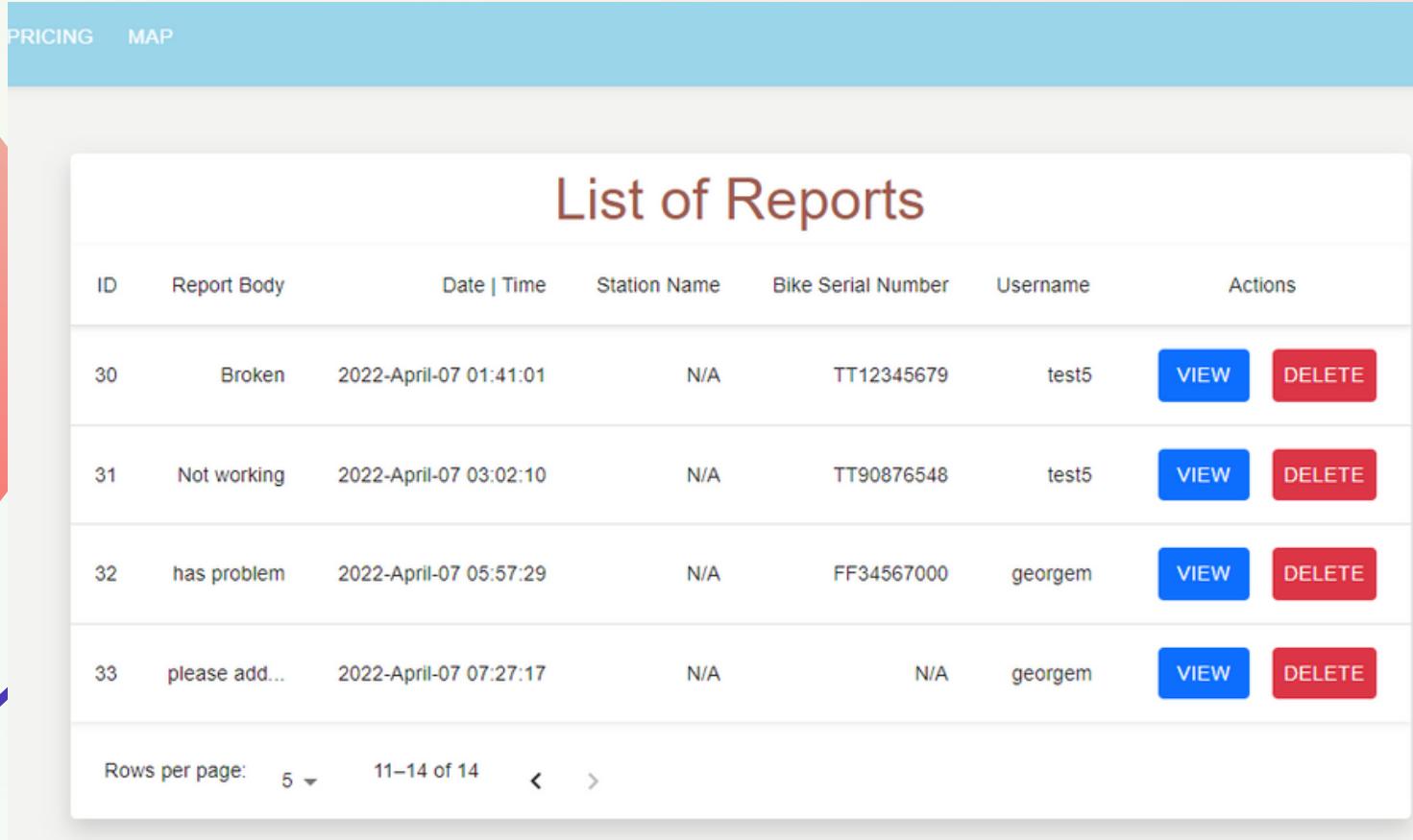
Start Station:
du Havre / de Rouen

RETURN BIKE



OUR SOLUTIONS: USER REPORTS

- Users can submit reports
 - Admins can view and delete reports





PRICING MAP



OUR SOLUTIONS:

BIKE MAINTENANCE

DateTime: 2022-April-02 09:14:15

User: jimmy

Station: Roy / St-Denis

Bike: AA12345678

Report Body:

s3 file


[Back to Reports](#) [Send Bike For Maintenance](#)

PRICING MAP

Bike is sent for maintenance sucessfully!

DateTime: 2022-April-07 09:14:15

User: jimmy

Station: Roy / St-Denis

Bike: AA12345678

Report Body:

s3 file

[Back to Reports](#)

PRICING MAP

Bike is maintained sucessfully!

List of Bikes In Maintenance

ID	Bike Type	Currently In Use	Last Maintenance	Serial Number	Status	Actions
2	STANDARD	false	2021-11-19	LO12345333	InMAINTENANCE	<button>MAINTAINED</button>
33	STANDARD	false	2022-03-28	TT90005000	InMAINTENANCE	<button>MAINTAINED</button>
34	STANDARD	false	2022-03-28	LL12345670	InMAINTENANCE	<button>MAINTAINED</button>
36	ELECTRIC	false	2022-03-29	KJ12345678	InMAINTENANCE	<button>MAINTAINED</button>
37	STANDARD	false	2022-04-01	LK09876543	InMAINTENANCE	<button>MAINTAINED</button>

Rows per page: 5 1-5 of 8 < >



PRICING MAP

Search here ...



DELETED BIKES

List of Bikes

ID▼	Bike Type	Currently In Use▼	Last Maintenance▼	Serial Number▲	Status	station Name	Actions
26	ELECTRIC	false	2022-04-07	AA12345678	ACTIVE	Warehouse	<button>EDIT!</button> <button>DELETE</button>
31	STANDARD	false	2022-04-07	LL12345678	ACTIVE	Warehouse	<button>EDIT!</button> <button>DELETE</button>
23	STANDARD	false	2022-04-07	TT90876548	ACTIVE	Warehouse	<button>EDIT!</button> <button>DELETE</button>
12	STANDARD	false	2022-04-06	SS12345678	ACTIVE	Roy / St-Denis	<button>EDIT!</button> <button>DELETE</button>

[ADD BIKE](#)

Showing Page 1 of 9

< < 1 > >

Copyright © Team KBAKT -2022.

this site is an imitation of the BIXI service for educational purposes



PRICING MAP

List of Bikes for Warehouse

ID	Serial Number	Bike Type	Last Maintenance Date	Currently In Use	Status	Transfer
23	TT90876548	STANDARD	2022-04-07	false	ACTIVE	<input type="checkbox"/>
26	AA12345678	ELECTRIC	2022-04-07	false	ACTIVE	<input type="checkbox"/>
31	LL12345678	STANDARD	2022-04-07	false	ACTIVE	<input type="checkbox"/>

Back to Stations TRANSFER BIKES Rows per page: 5 1-3 of 3 < >



PRICING MAP

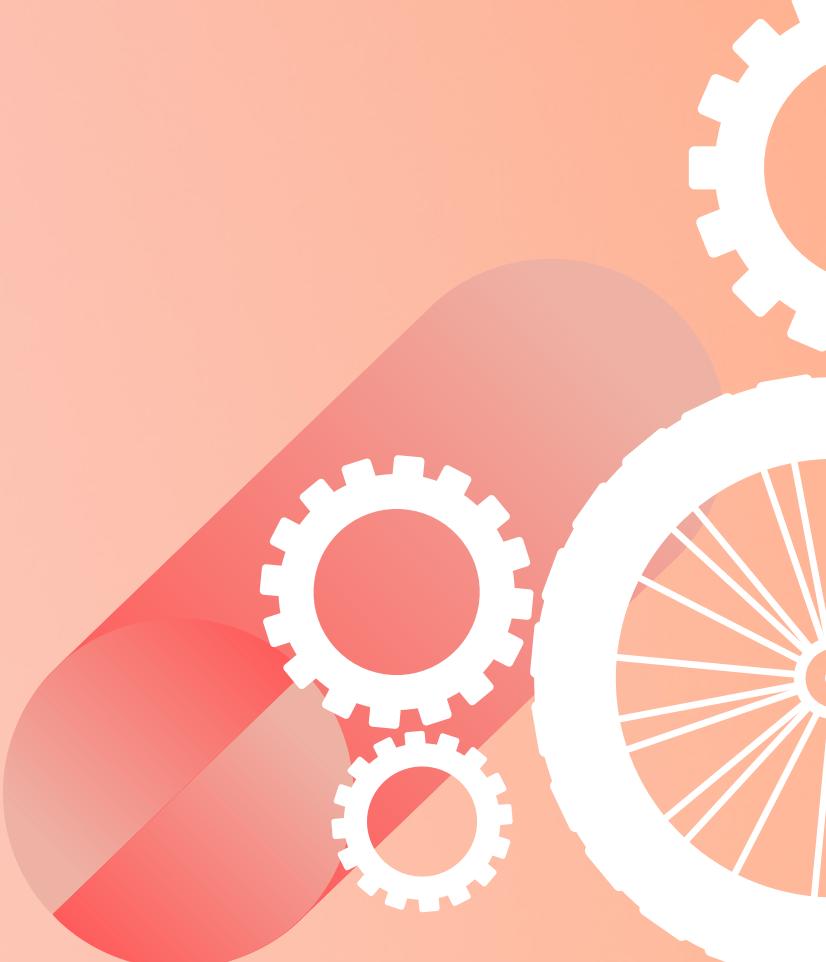
List of Bikes for Warehouse

ID	Serial Number	Bike Type	Last Maintenance Date	Currently In Use	Status	Transfer
23	TT90876548	STANDARD	2022-04-07	false	ACTIVE	<input checked="" type="checkbox"/>
26	AA12345678	ELECTRIC	2022-04-07	false	ACTIVE	<input checked="" type="checkbox"/>
31	LL12345678	STANDARD	2022-04-07	false	ACTIVE	<input type="checkbox"/>

Back to Stations TRANSFER BIKES There are not enough available slots at this station. Rows per page: 5 1-3 of 3 < >

OUR SOLUTION: ADMIN - BIKE TRANSFERS

- Admins can transfer bikes to different stations with available spots





PRICING

MAP



List of Users for Promos

ID	Username	Email	Last Name	First Name	Stripe Id	User Roles	Account Status	Transfer
30	potus1	POTUS@usa.com	Biden	Joe	cus_L07gOTmgz3Ymez	user	ACTIVE	<input type="checkbox"/>
31	DeerLady	jane@doe.com	Doe	Jane	cus_L07oGVWvfvcen	user	ACTIVE	<input type="checkbox"/>
32	jimmy	jimmy@fallon.com	Fallon	Jimmy	cus_L08B45H6YsIGm2	user	ACTIVE	<input type="checkbox"/>
33	negarina	negarina@negarina.com	negarina'	negarina		user	ACTIVE	<input type="checkbox"/>
34	kpost19	maggie@gmail.com	Post	Kathryn	cus_LP6kEeaWI6BovQ	user	ACTIVE	<input type="checkbox"/>

Promo: Expiration:

Rows per page: 11-15 of 71

Copyright © Team KBAKT -2022.

Admin Promos

Add Coupon

Add Coupon

Promo Title:

Choose Discount Type:

Percentage Off:

Expiration:

Copyright © Team KBAKT -2022.

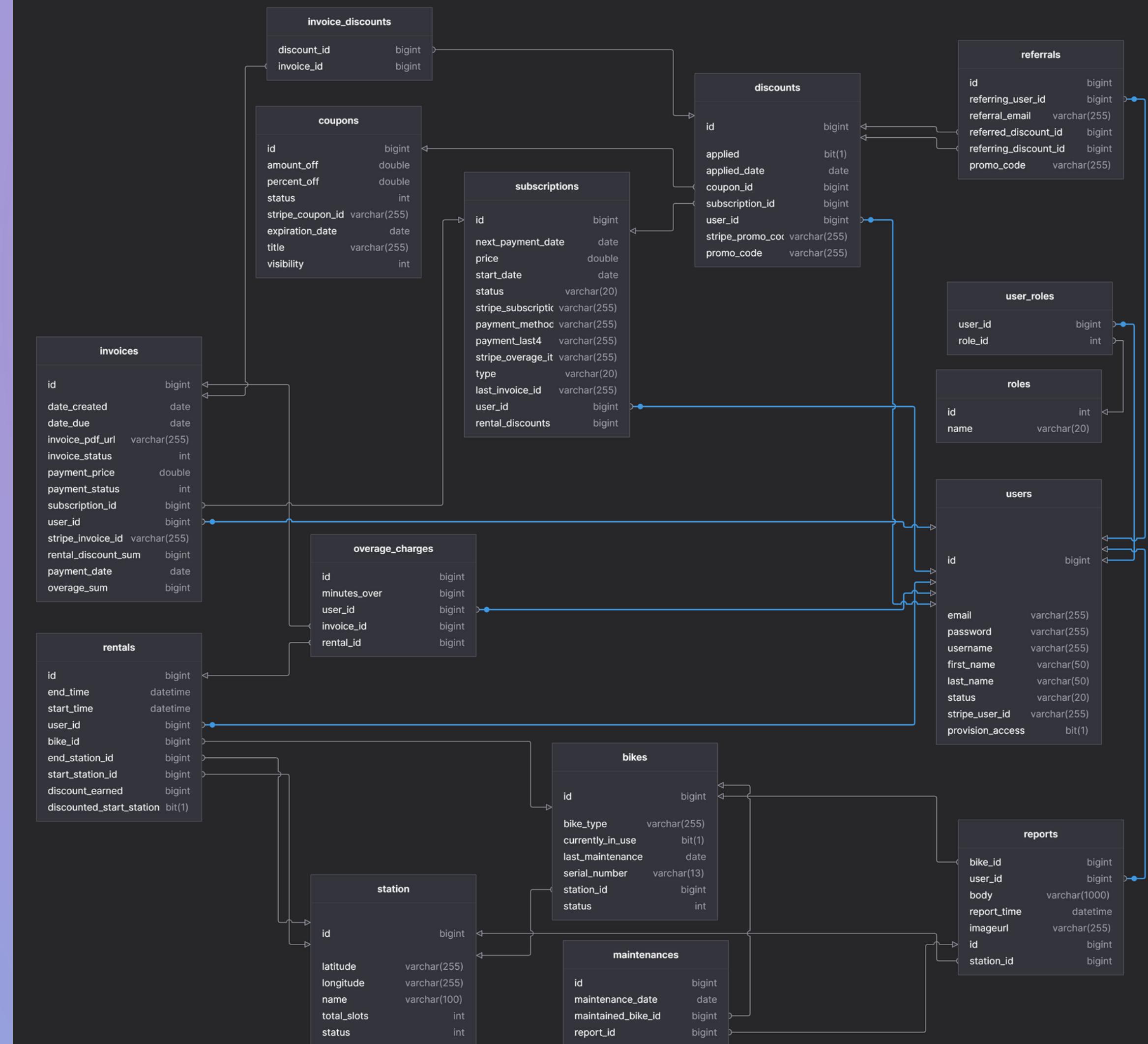
Don't miss out
Subscribe today and save!

Use this coupon to when you come back to us and save 50000\$ on this month's subscription plan. This coupon expires on 2022-04-06.

IBAIGOYW

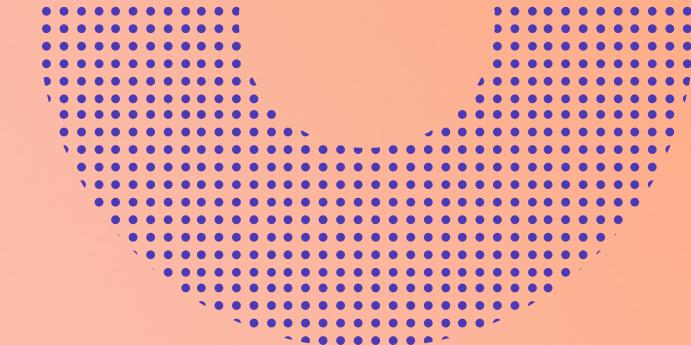


DATABASE DESIGN



MAPPING OUT YOUR RIDE

* Babel + ESLint + MapBox = Transpilation errors



Challenge: Rendering a MapBox element in a react app

The problem: React Apps use Babel to transpile the application "down" to ES5 but Mapbox require ES6



= Transpiler

Mapbox documentation => `import mapboxgl from '!mapbox-gl';`

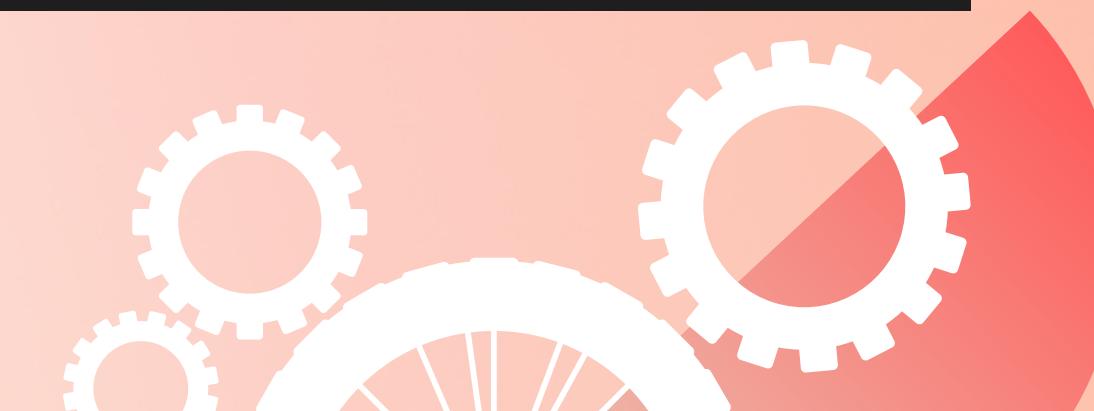
Solution:

-Use Webpack's inline loader syntax to inject the transpile prevention directly in the webpack build

```
mapboxgl.workerClass = require('worker-loader!mapbox-gl/dist/mapbox-gl-csp-worker').default;
```

-Tell ESLint to ignore this error so Create React App doesn't error out the build

```
import mapboxgl from '!mapbox-gl'; // eslint-disable-line import/no-webpack-loader-syntax
```



MAPPING OUT YOUR RIDE

* Searching for a station

Challenge: Searching for a station without re-rendering the map

The problem: submitting a traditional form via Post typically means refreshing or redirecting the page.

Solution: Leverage the power of react States!

```
const [postalSearch, setPostalSearch] = useState();
```

```
<input  
  type="search"  
  className="form-control rounded my-3 input-group rounded w-75"  
  placeholder="H1A 1A1"  
  aria-label="Search"  
  aria-describedby="Search"  
  onKeyUp={(event) => {  
    // postalCodeOnly(event.target.value);  
    setPostalSearch(event.target.value);  
  }}>  
  
<button  
  className="w-auto search-btn rounded-circle mx-4 shadow"  
  type="submit"  
  onClick={handleSearch}>
```

```
function handleSearch() {  
  let data = " ";  
  if (postalSearch !== null) {  
    data = { postalCode: postalSearch };  
  }  
  axios.post(`/api/stations/closestStation`, data).then((response) => {  
    let stationElementId = "link-" + response.data.id;  
    let stationClick = document.getElementById(stationElementId)  
    stationClick.click();  
  }).catch((error) => {  
    setError(error.response.data.error ?? error.response.data.message ?? "There was a problem");  
  });  
}
```

PAYING YOUR WAY

* Searching for a station

Challenge: Setting payment intent while keeping your secret safe.

The problem: Initially, our secret was stored on page load and a stripe ID was created regardless of success

PAYING YOUR WAY

* Searching for a station

```
import React, { useReducer } from 'react'; 7.4k (gzipped: 3k)

const StatusMessages = ({ messages }) =>
  messages.length ? (
    <div id="messages" role="alert">
      {messages.map((m, i) => (
        <div key={i}>{maybeLink(m)}</div>
      ))}
    </div>
  ) : (
    ''
  );

const maybeLink = (m) => {
  const piDashboardBase = 'https://dashboard.stripe.com/test/payments';
  return (
    <span dangerouslySetInnerHTML={{ __html: m.replace(
      /pi_(\S*)\b/g,
      `<a href="${piDashboardBase}/$1" target="_blank">$1</a>`)}></span>
  );
};

// Small hook for adding a message to a list of messages.
const useMessages = () => {
  // helper for displaying status messages.
  return useReducer((messages, message) => {
    // Embed link
    return [...messages, message];
  }, []);
};

export default StatusMessages;
export { useMessages };
```

```
const handleSubmit = async (e) => {
  if (accessToken) {
    e.preventDefault();
    if (!stripe || !elements) {
      return;
    }
    addMessage('Creating Payment Intent...');
    // API call to create payment intent with server.
    axios.post('/api/stripe/createSubscription', promoCode, {
      headers: {
        Authorization: 'Bearer ' + accessToken
      }
    }).then((response) => {
      const stripeClientSecret = response.data.clientSecret;
      addMessage('Payment Intent Created');
      const paymentIntent = stripe.confirmCardPayment(
        stripeClientSecret, {
          payment_method: {
            card: elements.getElement(CardElement),
          }
        }).then((result) => {
          if (result.error) {
            addMessage(result.error.message);
            return;
          }
          addMessage(`PaymentIntent ${paymentIntent.id}: ${paymentIntent.status}`);
          addMessage('Payment Success!');
          setTimeout(navigate('/registration/success'), 3000);
          // Successful subscription payment
        })
      .catch(err => {
        addMessage(`<span class="text-danger">${err.response.data}</span>`);
      });
    });
  }
}
```

Solution: Create the payment intent upon submission and set up messaging system to give real time feedback while we await webhook call from stripe.

SETTING UP GITHUB ACTIONS

CONFIGURATION FILE

HEROKU AUTOMATIC DEPLOY

 Automatic deploys from  main are enabled

Every push to `main` will deploy a new version of this app. Deploys happen automatically: be sure that this branch in GitHub is always in a deployable state and any tests have passed before you push. [Learn more](#).

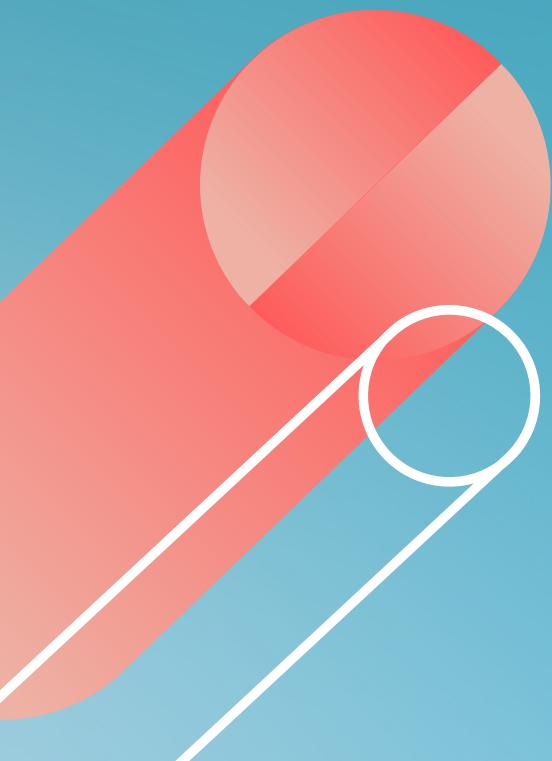
Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

[Disable Automatic Deploys](#)

```
3
4 name: Java CI with Maven
5
6 on:
7   push:
8     branches: [ main ]
9   pull_request:
10    branches: [ main ]
11
12 jobs:
13   build:
14
15     runs-on: ubuntu-latest
16
17   steps:
18     - uses: actions/checkout@v2
19     - name: Set up JDK 17
20       uses: actions/setup-java@v2
21       with:
22         java-version: '17'
23         distribution: 'temurin'
24         cache: 'maven'
25     - name: Build with Maven
26       env:
27         DB_URL: ${{ secrets.DB_URL }}
28         DB_USER_NAME: ${{ secrets.DB_USER_NAME }}
29         DB_PASSWORD: ${{ secrets.DB_PASSWORD }}
30         SECRET_KEY: ${{ secrets.SECRET_KEY }}
31         EXPIRE_TIME: 86400000
32       run: mvn -B package --file pom.xml
```

CI / CD PIPELINE



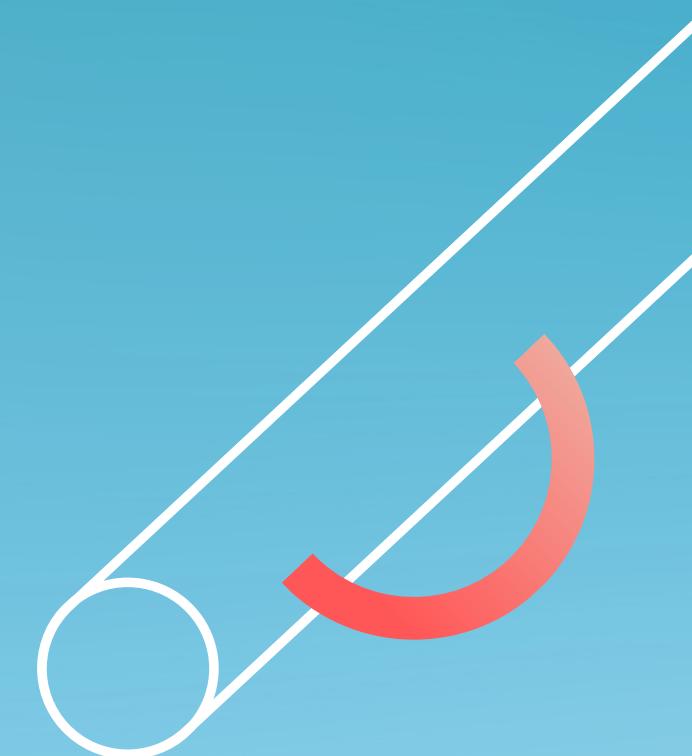
Front

VS-Code



Back

IntelliJ



React

```
11  public class SmokeTest {  
12  
13      @Autowired  
14      private AuthController authController;  
15  
16      @Autowired  
17      private BikeController bikeController;  
18  
19      @Autowired  
20      private StationController stationController;  
21  
22      @Autowired  
23      private StationUserController stationUserController;  
24  
25      @Autowired  
26      private TestController testController;
```

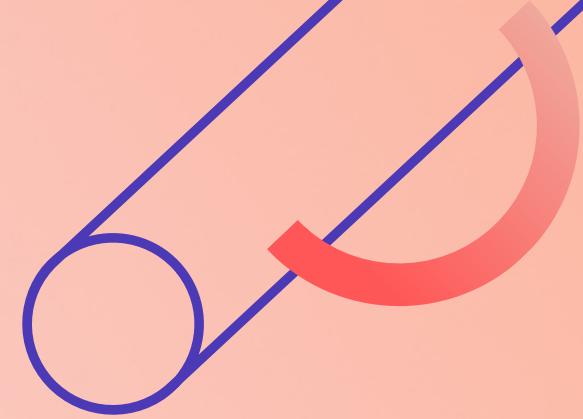
The screenshot shows a code editor interface with a dark theme. On the left, a file named `App.test.js` is open, containing the following code:

```
1 import { render } from '@testing-library/react';  
2 import Footer from './Components/Footer';  
3  
4 test('Footer renders without crashing', () => {  
5     | render(<Footer />);  
6 });  
7
```

Below the code editor, a terminal window displays the test results:

```
PROBLEMS 27 OUTPUT DEBUG CONSOLE COMMENTS  
▼ TERMINAL  
PASS src/App.test.js  
✓ Footer renders without crashing (53 ms)  
  
Test Suites: 1 passed, 1 total  
Tests: 1 passed, 1 total  
Snapshots: 0 total  
Time: 4.017 s  
Ran all test suites.  
  
Watch Usage: Press w to show more.[]
```

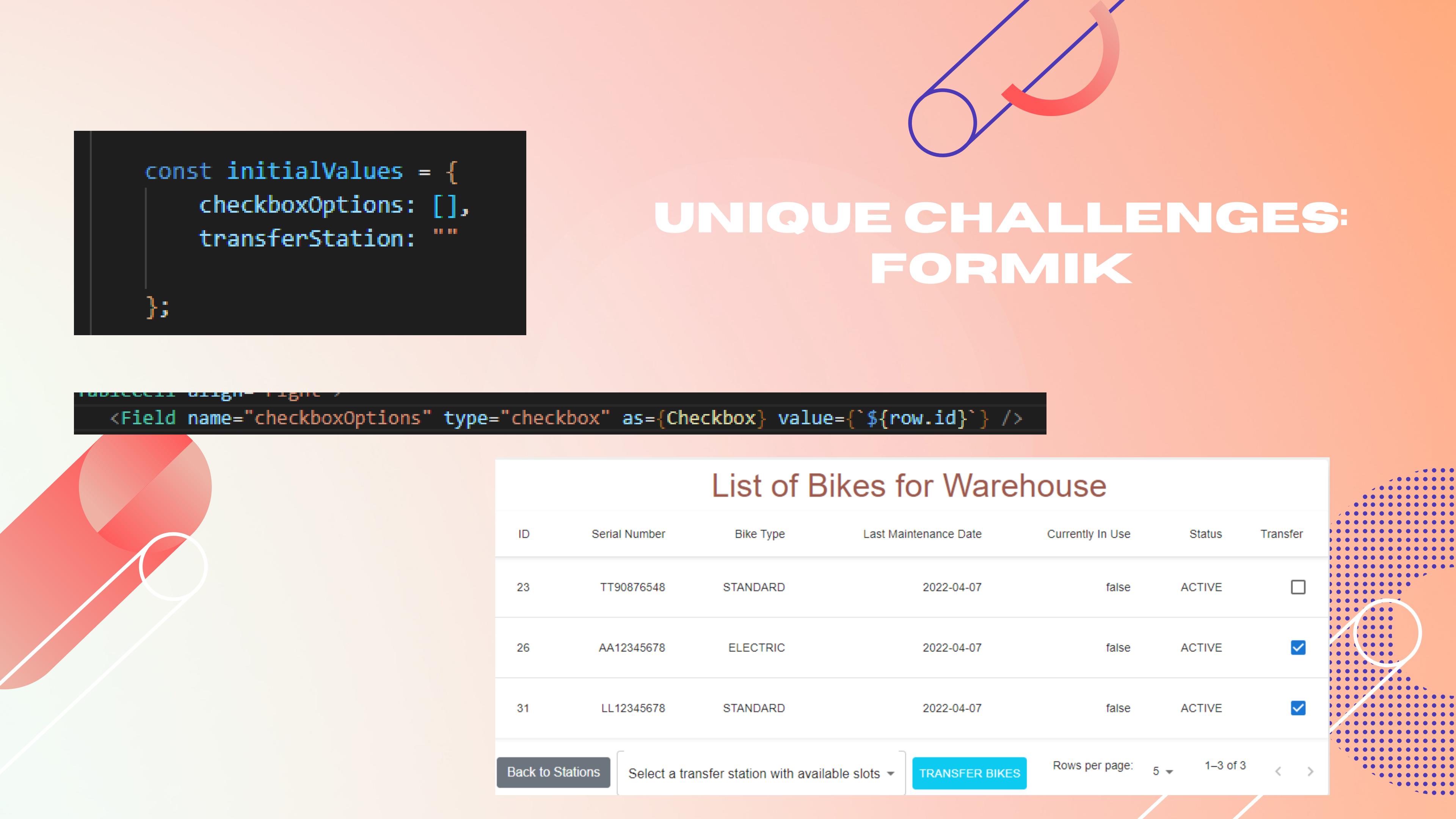
Spring



UNIQUE CHALLENGES: FORMIK

```
const initialValues = {  
  checkboxOptions: [],  
  transferStation: ""  
};
```

```
<Field name="checkboxOptions" type="checkbox" as={Checkbox} value={`${row.id}`} />
```



List of Bikes for Warehouse

ID	Serial Number	Bike Type	Last Maintenance Date	Currently In Use	Status	Transfer
23	TT90876548	STANDARD	2022-04-07	false	ACTIVE	<input type="checkbox"/>
26	AA12345678	ELECTRIC	2022-04-07	false	ACTIVE	<input checked="" type="checkbox"/>
31	LL12345678	STANDARD	2022-04-07	false	ACTIVE	<input checked="" type="checkbox"/>

[Back to Stations](#) Select a transfer station with available slots ▾ [TRANSFER BIKES](#) Rows per page: 5 1–3 of 3 < >

UNIQUE CHALLENGES: UPLOADING TO S3

```
<dependency>
    <groupId>io.awspring.cloud</groupId>
    <artifactId>spring-cloud-aws-context</artifactId>
    <version>2.4.0</version>
</dependency>
<dependency>
    <groupId>io.awspring.cloud</groupId>
    <artifactId>spring-cloud-aws-autoconfigure</artifactId>
    <version>2.4.0</version>
</dependency>
```



```
cloud.aws.credentials.access-key=${AWS_ACCESS_KEY}
cloud.aws.credentials.secret-key=${AWS_SECRET_KEY}
```



```
public void sendReport(UserReportRequest userReportRequest, MultipartFile file) throws IOException
```

```
String key = UUID.randomUUID() + "." + extension;
ObjectMetadata metaData = new ObjectMetadata();
metaData.setContentLength(file.getSize());
metaData.setContentType(file.getContentType());
awsS3Client.putObject(bucketName: "pyblog", key, file.getInputStream(), metaData);
```



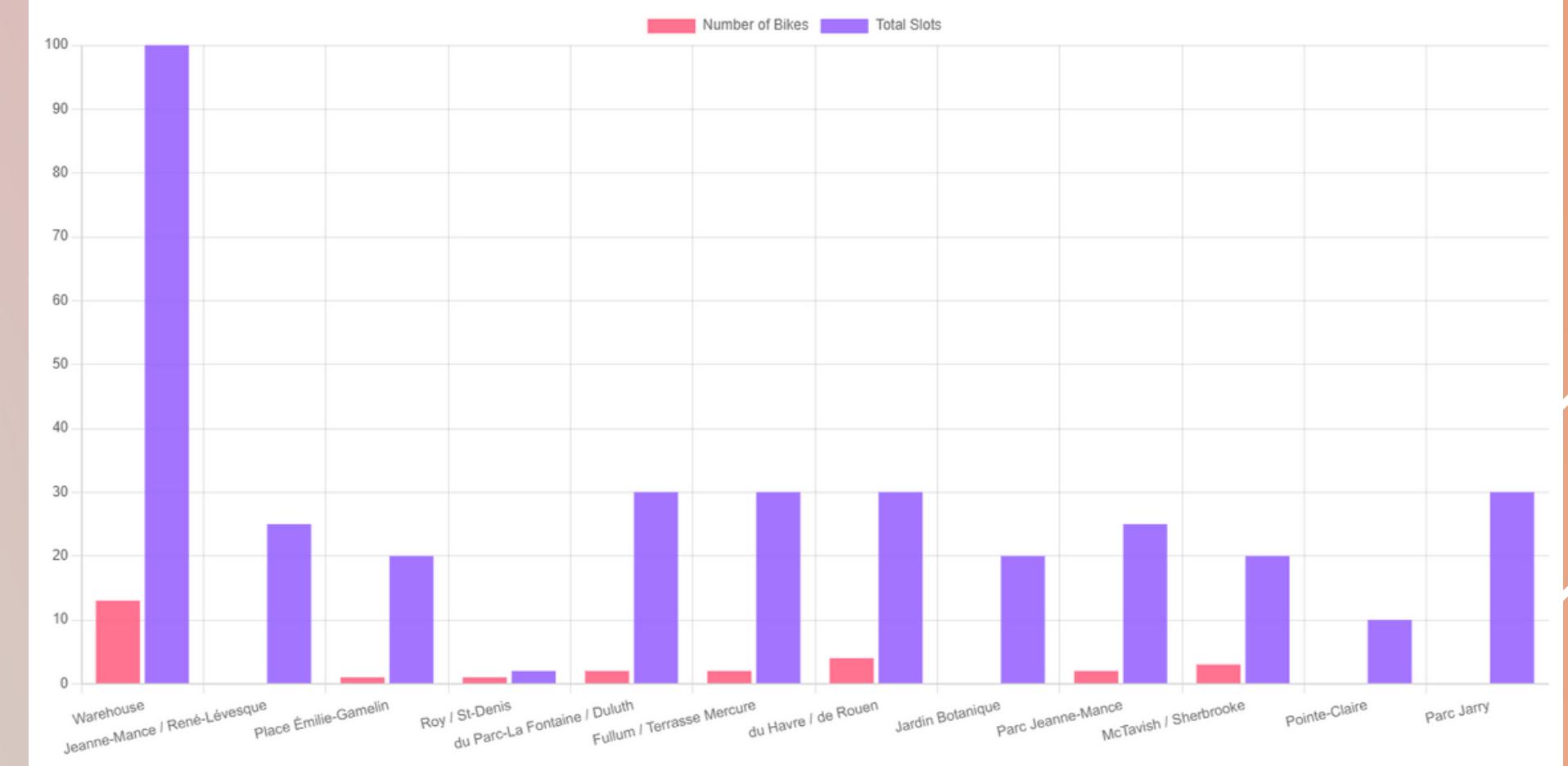
	Name	Type	Last modified	Size	Storage class
	1b5fb95-0ed8-4f7f-828d-6c09fb50e701.jpg	jpg	April 2, 2022, 11:45:28 (UTC-04:00)	258.7 KB	Standard

UNIQUE CHALLENGES: DATA DISPLAY

```
const [stationChartData, setStationChartData] = useState({})
```

```
setStationChartData({  
  
  labels: response.data.map((station) => station.name),  
  datasets: [  
    {  
      label: "Number of Bikes",  
      data: response.data.map((station) => station.numberOfBikes),  
      backgroundColor: [  
        'rgba(255, 99, 132, 0.9)',  
      ]  
    },  
    {  
      label: "Total Slots",  
      data: response.data.map((station) => station.totalSlots),  
      backgroundColor: ['rgba(153, 102, 255, 0.9)']  
    }  
  ]  
});
```

```
<div class="card-body">  
  {stationChartData.datasets && <BarChart chartData={stationChartData} />}  
</div>
```



CHALLENGE: GET MENU ITEM VALUES IN FORMIK

SOLUTION: CUSTOMIZEDSELECTFORFORMIK

```
const CustomizedSelectForFormik = ({ children, form, field }) => {
  const { name, value } = field;
  const { setFieldValue } = form;

  return (
    <Select
      name={name}
      value={value}
      onChange={e => {
        setFieldValue(name, e.target.value);
      }}
    >
      {children}
    </Select>
  );
};

<Field name="bikeType" component={CustomizedSelectForFormik} fullWidth>
  <MenuItem value={"STANDARD"} fullWidth>STANDARD</MenuItem>
  <MenuItem value={"ELECTRIC"} fullWidth>ELECTRIC</MenuItem>
</Field>
```

CHALLENGE: PREFILL REACT FORM TO UPDATE SOLUTION: HELPER.JS

```
export default function EditBikeHelper() {
  ...
  useEffect(() => {
    axios.get(`/api/admin/bikes/edit/${id}`, {
      headers: {
        Authorization: `Bearer ${localStorage.getItem('accessToken')}`
      }
    }).then((response) => {
      setData(response.data);
    });
  }, [id]);
  return data ? <EditBike preloadedValues={data}> : <div>Loading...</div>
}
```

```
export default function EditBike({preloadedValues}) {
  ...
  const {register, handleSubmit, formState: {errors} } = useForm({
    defaultValues: preloadedValues
  });
  ...
}
```

PAGINATION & SORTING & SEARCHING (BACKEND)

```
public interface BikeRepoPagination extends PagingAndSortingRepository<Bike, Long> {
```

```
@Query("SELECT b FROM Bike b LEFT JOIN b.station s WHERE b.status <> com.fsd.hellovelo.entity.EStatus.DISABLED" +  
    " AND CONCAT(b.serialNumber, ' ', b.bikeType) " +  
    " LIKE %:searchText% OR s.name LIKE %:searchText%")  
Page<Bike> findAll( Pageable pageable, @Param("searchText") String searchText);
```

```
public Page<Bike> findAllPaginatedBikeListWithSearch(int pageNumber, int pageSize, String sortBy, String sortDir, String searchText){  
  
    return bikeRepoPagination.findAll(  
        PageRequest.of(pageNumber, pageSize,  
            sortDir.equalsIgnoreCase( anotherString: "asc") ? Sort.by(sortBy).ascending() : Sort.by(sortBy).descending()  
        ), searchText  
    );
```

```
@GetMapping(PathVariable "/search/{searchText}")  
public Page<Bike> getAllPaginatedBikes( int pageNumber, int pageSize, String sortBy, String sortDir, @PathVariable String searchText) {  
    return bikeService.findAllPaginatedBikeListWithSearch( pageNumber,pageSize,sortBy,sortDir,searchText);  
}
```

PAGINATION & SORTING & SEARCHING (FRONT-END)

```
useEffect(() => {
  axios.get(`/api/admin/bikes/paginated?pageNumber=${currPage - 1}&pageSize=${rowsPerPage}&sortBy=id&sortDir=asc`, {
    headers: {
      Authorization: 'Bearer ' + localStorage.getItem('accessToken')
    }
  }).then((response) => {
    console.log(response.data)
    setListOfBikes(response.data.content);
    setBikeTotalPages(response.data.totalPages);
    setTotalBikeElements(response.data.totalElements);
    setCurrPage(response.data.number + 1);
  })
})
```

```
const updateTable = (currPage) => {
  let sortDir = sortDirection ? "asc" : "desc";
  axios.get(`/api/admin/bikes/paginated?pageNumber=${currPage - 1}&pageSize=${rowsPerPage}&sortBy=${sortField}&sortDir=${sortDir}`, {
    headers: {
      Authorization: 'Bearer ' + localStorage.getItem('accessToken')
    }
  }).then((response) => {
    setListOfBikes(response.data.content);
    setBikeTotalPages(response.data.totalPages);
    setTotalBikeElements(response.data.totalElements);
    setCurrPage(response.data.number + 1);
  })
}
```

```
const updateHandleSearch = (currPage) => {
  let sortDir = sortDirection ? "asc" : "desc";
  axios.get(`/api/admin/bikes/search/${searchText}?pageNumber=${currPage - 1}&pageSize=${rowsPerPage}&sortBy=${sortField}&sortDir=${sortDir}`, {
    headers: {
      Authorization: 'Bearer ' + localStorage.getItem('accessToken')
    }
  }).then((response) => {
    setListOfBikes(response.data.content);
    setBikeTotalPages(response.data.totalPages);
    setTotalBikeElements(response.data.totalElements);
    setCurrPage(response.data.number + 1);
  })
}
```

Webhooks

Hosted endpoints

URL

- 🌐 <https://helloworld.herokuapp.com/api/stripe/webhook/invoiceCreated>
- 🌐 <https://helloworld.herokuapp.com/api/stripe/webhook/provisionAccess>
- 🌐 <https://helloworld.herokuapp.com/api/stripe/webhook/setDefaultPayment>
- 🌐 <https://helloworld.herokuapp.com/api/stripe/webhook/invoiceStatus>

CHALLENGES: STRIPE WEBHOOKS

```
@PostMapping("/webhook/invoiceCreated")
public ResponseEntity<?> addDiscounts(@RequestBody String json, @RequestHeader("Stripe-Signature") String sigHeader){
    try{
        Event event = StripeUtility.getEvent(json, sigHeader, System.getenv("INVOICE_CREATED_KEY"));
        StripeObject stripeObject = StripeUtility.getStripeObject(event);
    }
}
```

```
public static Event getEvent(String json, String sigHeader, String secret) {
    try {
        return Webhook.constructEvent(json, sigHeader, secret);
    } catch (SignatureVerificationException e) {
        // Invalid signature
        throw new IllegalArgumentException(json + "\n" + sigHeader + "\n" + secret);
    }
}
```

```
Subscription subscription =
    Subscription.create(subCreateParams);

com.fsd.hellovelo.entity.Subscription localSubscription = new com.fsd.hellovelo.entity.Subscription(LocalD
localSubscription.setStripeSubscriptionId(subscription.getId());
localSubscription.setPrice(subscription.getItems().getData().get(0).getPrice().getUnitAmountDecimal());

subscriptionRepository.saveAndFlush(localSubscription);
```

⌚ invoice.created	11:38:49 p.m.
✖ invoice.created	11:32:50 p.m.
✖ invoice.created	11:30:50 p.m.
✖ invoice.created	11:30:17 p.m.
✖ invoice.created	11:26:21 p.m.
✖ invoice.created	11:13:40 p.m.
✖ invoice.created	11:04:07 p.m.
✖ invoice.created	11:01:23 p.m.
✖ invoice.created	10:42:04 p.m.
✖ invoice.created	10:33:31 p.m.
✖ invoice.created	10:30:06 p.m.

SAVE AND FLUSH

Failed webhooks due to async webhooks
and repository flushing

POSTAL CODE SEARCH

Step 1: Validation

```
String regex = "[ABCEGHJ-NPRSTVXYabceghj-nprstvxy]\\d[ABCEGHJ-NPRSTV-Zabceghj-nprstv-z][-]?\\d[ABCEGHJ-NPRSTV-Zabceghj-nprstv-z]\\d";
if(!postalCode.getPostalCode().matches(regex)){
    throw new ResourceNotFoundException("Error: Invalid postal code.");
}
```

Step 2: Geocoding

```
String url = "http://www.mapquestapi.com/geocoding/v1/address?key=" + System.getenv("GEOCODE_KEY") + "&location=" + postalCode;

RestTemplate restTemplate = new RestTemplate();
GeocodeResponse value = restTemplate.getForObject(url, GeocodeResponse.class);
if(value == null){
    throw new ResourceNotFoundException("Error: Postal Code could not be found.");
}
```

Step 3: Distance comparison

```
private static final double r2d = 180.0D / 3.141592653589793D;
private static final double d2r = 3.141592653589793D / 180.0D;
private static final double d2km = 111189.57696D * r2d;
public static double meters(double lt1, double ln1, double lt2, double ln2) {
    double x = lt1 * d2r;
    double y = lt2 * d2r;
    return Math.acos( Math.sin(x) * Math.sin(y) + Math.cos(x) * Math.cos(y) * Math.cos(d2r * (ln1 - ln2))) * d2km;
}
```

HV
Hello Velo
Wed 2022-04-06 2:49 AM
To: Post, Kathryn

Don't miss out

You've been referred
to Hello Velo!

Use this coupon to save \$5 on this month's subscription plan.

Code: NWO5GJO9

Subscribe to Hello Velo



Referrals

Enter the email for your referral:

kate@post.com

Refer a Friend



Register Payment Info

Card Info

4242 04 / 24 242 42424

Promo Code

NWO5GJO9

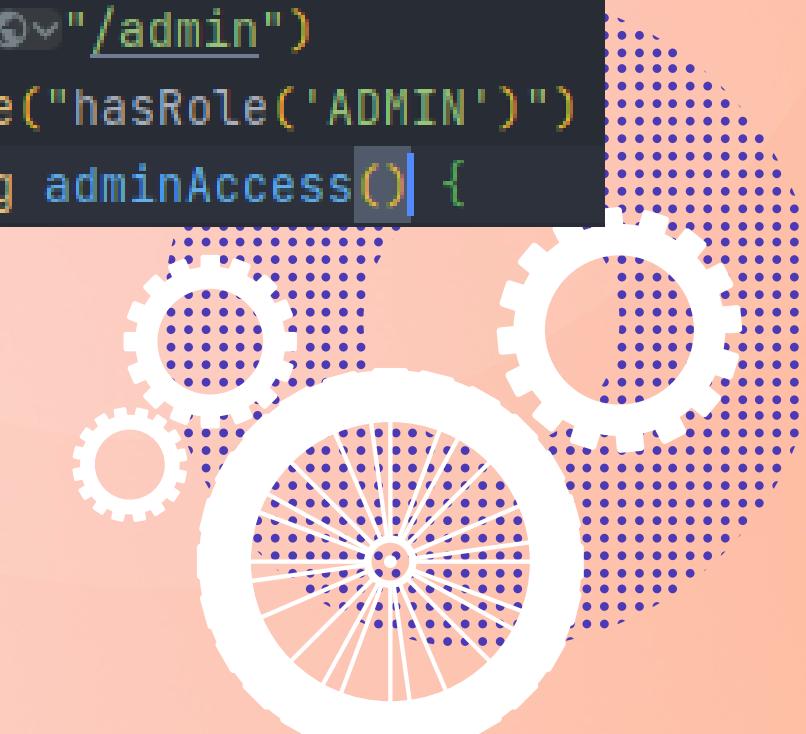
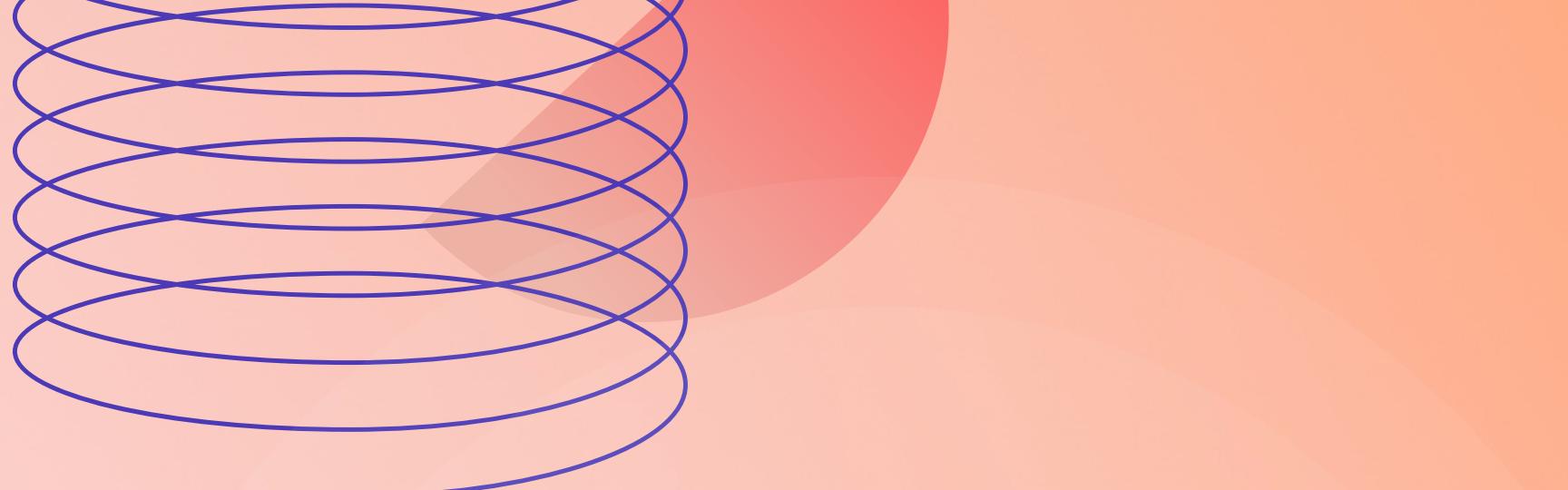
REGISTER CARD

AUTHENTICATING USERS IN SPRING BOOT WITHOUT THYMELEAF

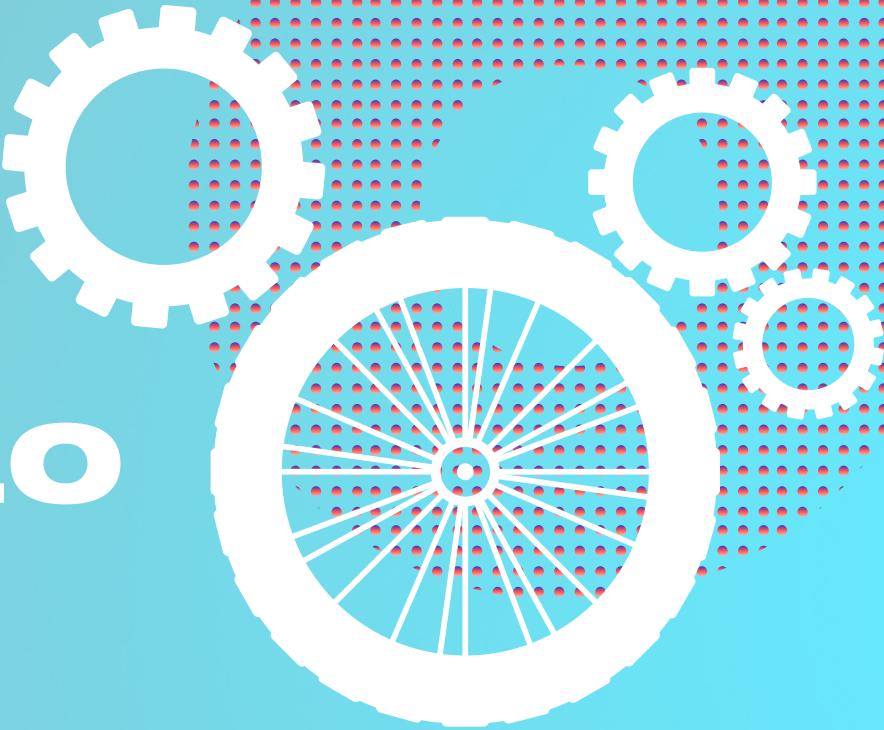
```
@Value("${SECRET_KEY}")
private String jwtSecret;
@Value("${EXPIRE_TIME}")
private int jwtExpirationMs;
public String generateJwtToken(Authentication authentication) {
    UserDetailsService userPrincipal = (UserDetailsService) authentication.getPrincipal();
    return Jwts.builder()
        .setSubject(userPrincipal.getUsername())
        .setIssuedAt(new Date())
        .setExpiration(new Date((new Date()).getTime() + jwtExpirationMs))
        .signWith(SignatureAlgorithm.HS512, jwtSecret)
        .compact();
}
public String getUserNameFromJwtToken(String token) {
    return Jwts.parser().setSigningKey(jwtSecret).parseClaimsJws(token).getBody().getSubject();
}
public boolean validateJwtToken(String authToken) {
    try {
        Jwts.parser().setSigningKey(jwtSecret).parseClaimsJws(authToken);
        return true;
    } catch (SignatureException e) {
```

```
@CrossOrigin(origins = "*", maxAge = 3600)
@RestController
@PreAuthorize("hasRole('ADMIN')")
@RequestMapping("/api/admin/reports")
public class ReportController {

    @GetMapping("/admin")
    @PreAuthorize("hasRole('ADMIN')")
    public String adminAccess() {
```



THE FUTURE OF HELLO VELO



QR CODES

Riders can scan their bike to process their rental



PAY PER USE

Non-subscribed customers can rent bikes too



CO2 OFFSET

View your contribution to the preservation of our global ecosystem



CARS

Hello-velo would like to offer car rental to our customers



SUMMARY

Accomplished many
of our goals

Enhanced our
knowledge of React.js
& Java Spring Boot

Learned new tech
(e.g. Jira, Git
branching)

Worked hard & had
some fun!