**Optimize static assets of contact form application**

**Context**

Prime Solutions is an IT solutions provider company who has earned reputation in providing IT solutions in different business domains.​​​

Recently they have been approached to develop OneStopSolutions web application.​

Through this application the consumers can reach out to various services.​​​

One of the key component of the app is to build a Contact book for all the service providers.​​​

Prime Solutions have approached Graffitti designers for a good UI design for managing contacts..​​​

The design is ready and functional.​​

Contact Form

Graffitti Designers have designed contact form that allows app users to provide inputs

**Challenge**

In urban cities, OneStopSolutions web application has earned a good popularity. ​

However, users in these cities often tend to access application via mobile devices.​

Hence, it is critical to ensure that the user experience is not compromised on the mobile devices​

Mobile devices often face challenges with low network bandwidth​

So, for smoother UX to mobile users bundling will help reduce latency and help with dead code elimination whereby ensuring there is no dip in the performance of the application​

**Problem Statement**

Minify and bundle the scripts code used by Contacts form using the module bundler tool - webpack

The bundled output should comprise of bundled JavaScript codes, index.html and stylesheet codes

Stylesheet should be reference by index.html using the tag and should not be embedded

The code may get modified with enhanced features which will eventually result into modifying the bundled code.

Implement cache busting so that the browser always loads modified scripts rather than loading from the cache.

**Instructions**

1. Download and unzip the boilerplate code.
2. Run the command **npm install** to install the dependencies required for automated testing.
3. Open the boilerplate code in VSCode to develop the assignment solution.
4. Copy the Contact form solution created in previous sprint in the cloned repository.
5. Install the required dependencies for bundling.
6. Create configuration file for webpack.
7. Run the command to create bundles.
8. Test the bundled code and ensure Contact form runs successfully.
9. Run the command **npm run test** to test the code for verifying existence of webpack configuration options.
10. Refactor the solution to ensure all test cases are passing.
11. DO NOT MODIFY THE PROVIDED CODE, ELSE THIS MAY IMPACT THE TEST CODE EXECUTION.
12. Zip the solution code by selecting all the files and folders **excluding the node\_modules folder** and give the name same as assignment name to the zipped file.
13. Upload the zipped solution for submission.