We also instantiate the DecPark using a PC server as CS and one android smartphone as a driver, two Android virtual machines as testing devices.

Specially, we use Geth (Go-Ethereum Client) as the primary tool for Ethereum network environment establishing. We instantiate four SPSPS on four nodes on the PC server using VMware Virtual Machine and Geth, each node has a signer account with authority to seal a block. The CS uses Spring Boot projects at IDEA. Http is used for communication between the smartphone and the server-side. We create the application on the Android smartphone and use Gradle to introduce the Web3J library to interact with the consortium blockchain. We use the javax.crypto library to implement the cryptographic primitives. We store the data generated during communication in a database using SQLite The screenshots of the application are shown in Fig. 7: a driver must submit his information to register to the CS and CB(Fig. 7(a)). After logging into the park querying system (Fig. 7(b)), a driver need interact with one of the SPSPS to complete anonymously authentication (Fig. 7(c)). By inputting the parking lot identity and query parameter, the authenticated driver can query the previously matched parking lot, the location of the returned results is marked on the map (Fig. 7(d)), and the matched parking lot can be reserved(Fig7 .(e)).