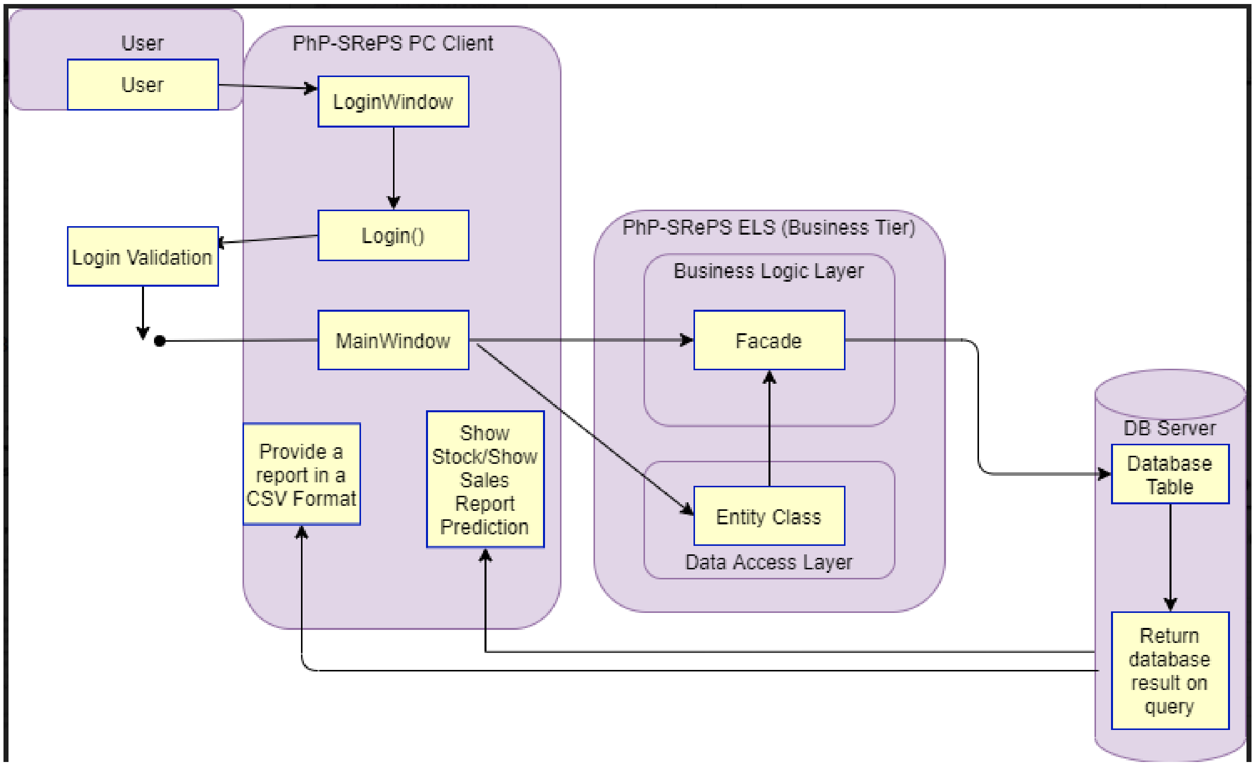
TASK - 22P

PHP-SRePS

Sales Reporting and Prediction System

GROUP MEMBERS	STUDENT IDENTIFICATION
[redacted]	[redacted]
[redacted]	[redacted]
[redacted]	[redacted]
Marcus Rakkhit	[redacted]
[redacted]	[redacted]

Tutorial - Monday 2:30pm EN402 with [redacted]



Overall our design did not differ between the 1st and 2nd sprint. This has remained as a good design that follows certain principles of a design. All the CRUD operations within the code remain in the facade that is kept in the Business layer, which is an example of high cohesion. This high cohesion keeps all the CRUD operations to be dealt with on only the database side of the program, while the results are returned to the issue in various ways. This is displayed by the connection between the Database Table and the Facade. The classes within this program also don’t rely on other classes too much for them to complete their normal day to day actions. This is an example of low coupling, showing that if one class begins to have issues it won’t bring other classes down with it. Our design that we have created is easy to follow and execute, meaning that it fulfils the requirements needed for high-level design philosophy.