Steps involved to build a Large PHP Application

If you are a layman in software development, here are the steps for you to build a large application for your company. These are best when implemented on an **agile** basis. It still may contain some technical jargon, we will be happy to clarify it if requested ...

1. **Understand the basic concept and have a high level requirement** document that will describe the business objective, overall concept and capabilities. List out all the possible features under each capabilities.
2. **Identify the 20% main features** that basically constitute 80% product values. Put them in the MVP and pick the 1st feature for development.
3. **Break down the feature into user stories**. In the proposed software who will do what, when, where and why. You need a **Product Owner** (it is an agile software development term) to efficiently do above jobs .
4. **Plan to convert each user story into software components** along with requirements and specifications associated with every future component. Components are user Interfaces and code objects, database objects etc - bricks and mortars of the software application. This job is of a **System Analyst cum Designer**.
5. **Create graphical representation** of the user interface layouts and screen templates. You need a **Graphics Designer** who can do it efficiently.
6. **Convert all the functionalities, quality benchmarks, environment setup activities, miscellaneous works into doable tasks** in a manner such that every one in the team (including you people) transparently see the progress, spent hours, estimates, issues etc. This is best done by putting all requirements, defects, test cases, general tasks etc on an integrated project management software. The person who manages it is called **Scrum Master**.
7. **Convert the proposed components along with requirements into actual software components**. Actual codes, files and database tables that can run on a software platform to provide expected results. This is ideally done by **Developer**. It's often done by two developers. Front end developer (**ReactJS, React Native Developer**) for creating user interfaces and **PHP/Node.js API** developer to fetch data from the server.
8. **Test (and fix) the React Native app / feature** formally using a test management tool to verify and validate the functional and nonfunctional TDD points are implemented. It requires further testing whether the idea itself requires changes or not. It may also require testing against load and security. Finally testing and fixing to make sure that everything is usable by the final user. This job is of a **Test Analyst, Tester and Product Owner**.
9. **Test the feature by the product owner and operation team** (customer team). Business facing testing to ensure all the business requirements meet or not. Exploratory testing to see if there are rooms for improvement. Usability testing by the actual system user and make sure they feel comfortable.
10. **Make the feature live** as per live process checkpoint. Everyone connected to the development team gets involved in the live process.

The entire cycle is done on an **agile process**. That means full development is transparent to the DEV team and Operation team. Right person does his job when required. An efficient feedback loop is created. If a defect /anomaly is identified at any step, it is sent back to the previous step and gets

corrected. Using this **LEAN / DevOps** powered Agile development process we can build a system which will provide maximum value to your users without delivering clutter.