Лабораторная работа №3
на тему : «Исследование архитектурного решения»
Ход работы:
Часть 1.
Проектирование архитектуры
Для разрабатываемой системы на высоком уровне абстракций:
1. Тип приложения: Веб-приложение для выполнения преимущественно на сервере в сценариях с постоянным подключением и приложение - клиент для мобильных устройств на базе ОС Android. Реализуется клиент-серверная архитектура
2. Стратегия распределенного развертывания
3. Обосновать выбор технологии
В нашем проекте используются следующие технологии: Laravel, VueJS, JWT. Выбор пал на них, т.к. уже имеется опыт работы с этими технологиями, они обеспечивают высокую производительность и надёжность системы. А так же для каждой из этих технологий имеется обширная документация и поддержка интернет сообщества по возникающим вопросам. Немаловажным аспектом является и то, что они идеально подходят для нашей клиент-серверной архитектуры и веб - приложения.
4. Показатели качества: безопасность, централизованный доступ к данным, простота обслуживания, надёжность, обеспеченность технической поддержкой, тестируемость.
5. Обозначить пути реализации сквозной функциональности.
Аутентификация и авторизация для обеспечения конфиденциальности пользовательских данных и надёжной работы системы в целом, а также связь между компонентами системы посредством API.
6. Наша “Архитектура To Be”
Приведены диаграммы компонентов “Component view.png” и развёртывания “Deployment view.png”.

Часть 2.
1. Анализ архитектуры разрабатываемого приложения
В данном проекте мы придерживаемся клиент-серверной архитектуры и стратегии нераспределённого развёртывания. Пытаемся соответствовать заданным критериям качества, таким как надёжность, централизованный доступ к данным и др.
2. Обобщённое представление архитектуры полность совпадает с приведённой в предыдущей части и приведена на тех же рисунках: “Component view.png” и “Deployment view.png”.
3. Архитектура «As is»
Диаграмма классов приведена на рисунке “Class diagram.jpg”

Часть 3.
Сравнение и рефакторинг
1. Сравнить архитектуры «As is» и «To be», выделить отличия и проанализировать их причины:
Мы стараемся полностью придерживаться изначально выбранной клиент-серверной архитектуры, однако мы решили перейти от распределённой стратегии развёртывания к нераспределённой. причиной тому послужило то, что мы решили проводить всю обработку данных на сервере и наше android - приложение является лишь графической оболочкой (удалённым терминалом) интерфейса, реализуемого в веб- приложении. Это позволило уйти от дублирования функциональности и снизить издержки на разработку, а также позволит легко и удобно разрабатывать приложения под другие платформы.
3. Пути улучшения архитектуры
Первым путём можно считать единообразие шаблонов в рамках одного слоя,

во-вторых мы уже придерживаемся и будем дальше придерживаться такой практики проектирования как избежание дублирования функциональности. В-третьих мы придерживаемся и будем придерживаться единого стиля кода и наименования. В дальнейшем мы будем стараться придерживаться принципов разделения функций (high cohesion & low coupling), единой ответственности и минимального знания.