Я хотел бы представить вашему вниманию курсовую работу на тему: «Разработка системы учета и контроля рабочего времени ИТ-компании». Курсовая работа состоит из двух глав, основная часть, и специальная. Первая часть посвящена теоретической составляющей, а специальная посвящена описанию алгоритма программы, работы с базой данных, а так же описанию пользовательского интерфейса системы.

Целью данной работы является исследование подхода к исчислению рабочего времени программиста, так же рассмотрение с практической точки зрения удобство разных подходов их преимущества и недостатки. Эта тема – актуальна, так как в современной разработке программного обеспечения важную роль играет правильная оценка времени на реализацию проекта. В данный момент существует множество систем учета рабочего времени, все они имеют свои преимущества и недостатки. В большинстве из них очень много лишнего и избыточного функционала, который только отвлекает от быстрого и удобного использования в повседневной жизни. Такие системы обычно используют подход логирования или тайм-трекинга времени. Как показывает практика, эти подходы имеют свои недостатки. Первый, связанный с логированием, чаще всего программисты завышают свои оценки или вовсе забывают указать отработанное время, после чего восстанавливая его из памяти, оценивают его с большими погрешностями. Второй недостаток связан с тайм-трекингом, чаще всего программист работает над несколькими задачами одновременно или использует парное программирование, включив таймер на одной задаче, он может переключиться на другую, что снова приводит к большим погрешностям

Все эти недостатки ведут в будущем к неправильному планированию, и ошибках в сроках разработки программных продуктов, что в свою очередь ведет к большим финансовым потерям. Так же менеджер проекта при планировании должен видеть реальную картину временных рамок, на которую нужно ориентироваться при разработке бизнес модели проекта, он должен знать точные временные затраты у каждого отдельно взятого программиста, это приводит к минимизации рисков на проектах. При правильном учете времени работника мы можем знать примерное время, которое ему нужно на реализацию той или иной задачи.

Я разработал алгоритм в основу которого легло объединение двух самых распространённых алгоритмов хронометража. Первый это логирование времени при котором работник сам указывает отработанные часы для каждой задачи, что ведет ко многим неточностям. Второй подход тайм-трекинг, при котором работник запускает таймер на каждой отдельной задаче, он имеет больше преимуществ в точности, но не исключает, что программист может переключиться на более приоритетную задачу и таким образом вызовет большие неточности в потраченном времени. Созданный подход должен минимизировать погрешности, он заключается в том, что работник накапливает время в так называемый банк времени с помощью трекера который не привязан к задачам синхронизированных из системы YouTrack. Система YouTrack тесно вплетена в разработанную программу пользователь для того что бы открыть часть функционала в системе должен иметь активный аккаунт, и синхронизировать его в настройках программы. Она была выбрана из множества систем по таким критериям как удобство интерфейса пользователя, а так же хорошо документрированное REST API, вся статистика компании собирается в ней, но вот личную собирает реализованная система.

Реализованная программа представляет собой Web-приложение на современном фремфорке laravel который позволяет создавать быстрые красивые ООП архитектуры. Фреймворк использует базы данных MySQL, но для работы с ней, он реализует так называемые фабрики, которые представляют собой часть большого объекта DB, и позволяют строить SQL запросы автоматически

После регистрации пользователю доступна только часть функционала это тайм-трекер и раздел статистики. На котором мы видим график средней активности за каждый взяты, конкретны день, а так же историю за все время. После того как пользователь введет на странице настроек свои доступы к системе YouTrack автоматически с помощью REST API синхронизируются данные по всем проекта, а так же задачи со всей информацией о них. Эти данные сохраняются в базу данных, после чего при каждом запросе, данные из системы сверяют даты, и определят, изменились данные на сервере и нужно ли их обновить, таким образом, в системе постоянно остаются актуальными. На странице трекера программист может начать исчисление времени, после остановки данные синхронизируются и добавляются в общий банк времени, из него на странице задачи пользователь может перевести часть времени на сервер YouTrack. Таким образом мы сохраняем все данные о времени на каждую задачу, и времени которое отработал работник.

В ходе проделанной работы я исследовал методы исчисления рабочего времени, а так же разработал систему на основе сервиса YouTrack, где реализовал собственный алгоритм хронометража.