Федеральное государственное образовательное бюджетное учреждение высшего образования "Финансовый университет при Правительстве Российской Федерации"

**Колледж информатики и программирования**

**Д Н Е В Н И К**

**производственной (профессиональной) практики**

Студента Деменчука Георгия Максимовича

Группа 4ПКС-115

Специальность 09.02.03 «Программирование в компьютерных системах»

**Москва**

**2019**

**Записи о работах, выполненных на практике**

**Промежуточные результаты производственной практики.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Дата | | **ПРОИЗВОДСТВЕННАЯ ПРАКТИКА ПО ПРОФИЛЮ СПЕЦИАЛЬНОСТИ**  Краткое содержание выполненных работ по вопросам раздела (темы), программы практики | Подписи руководителей | |
| от предприятия | от колледжа |
| 12.01.19 – 15.02.19 | *14.01.19* | Изучение теории по СУБД и реляционные СУБД, типы этих СУБД, изучение движков таблиц MySQL, выбор InnoDB. |  |  |
| 15.*01.19* | Разбор механизма транзакций в движках БД, разбор уровней блокировки, агрегатные функции. |  |  |
| 16.01.19 | Подробный разбор конкурентных запросов и механизма взаимных блокировок, способы избежания deadlock’ов. |  |  |
| 17.01.19 | Установка пакетов MySQL, Apache, PHP и Docker на дистрибутив CentOS: добавление нового пользователя, конфигурирование локальной СУБД на Mac OS для оптимизации времени дальнейшей отладки ПО. |  |  |
| 18.01.19 | Проектирование полей, таблиц и их взаимодействия в БД, изучение и сравнение инструмента MySQL Workbench с PHPMyAdmin, создание схемы БД. |  |  |
| 21.01.19 | Создание БД по ранее утвержденной схеме, небольшие правки относительно типов полей БД. |  |  |
| 22.01.19-23.01.19 | Изучение теории, связанной с ORM, написание собственного ORM-скрипта на Python для заполнения БД тестовыми данными, знакомство с PonyORM, практическое ознакомление с возможностью применения SQL -инъекций из-за использования строковой интерполяции в формирования запроса на клиенте, исправление возникшего недочета. |  |  |
| 24.01.19-25.01.19 | Изучение YAML для хранение данных конфигурации, написание программы-клиента для заполнения таблицы Code для хранения данных кодов строк  в формах бухгалтерской отчетности, указываемые в годовой  бухгалтерской отчетности организации, представляемой  в органы государственной статистики и другие органы  исполнительной власти. |  |  |
| 28.01.19 | Реализация механизма транзакций для программы заполнения БД тестовыми данными, общая реструктуризация, нагрузочное тестирование, запись финальных тестовых данных в БД, начало создания материалов для презентации к защите по модулю ПМ.02. |  |  |
| 29.01.19 | Принятие решения об использовании в docker-контейнере РСУБ класса NOSQL Redis для хранения очереди обрабатываемых данных. Также в docker в качестве REST WEB API будет использоваться Flask. |  |  |
| 30.01.19-1.02.19 | Написание концепта Flask-приложения и DockerImage для понимания архитектуры проекта, практическое знакомство с модулем Redis для Python3, а также контейнерами redislabs/redis-py, python:3.7, redis:5.0.3. |  |  |
| 4.02.19-6.02.19 | Развертка контейнера LAMP в docker, перенос данных из локальной СУБД MySQL в контейнер, использование docker-composer. |  |  |
| *7.02.19* | Структуризация материалов для защиты отчета по практике, дозаполнение отчета по производственной практике по модулю ПМ.02, оформление дневника практики. |  |  |
| *8.02.19-15.02.19* | Создание презентации к защите отчета по производственной практике, получение необходимых подписей для отчетов. |  |  |
| Результат производственной практики по модулю ПМ.02 «Разработка и администрирование баз данных» | | | Оценка (зачет/незачет) | Подпись руководителя от предприятия (М.П.) |
|  |  |
| 16.02.19 – 22.03.19 | *16.02.19* | Анализ возможных вариантов алгоритмов для определения типа документа (формы 01 и 02) |  |  |
| *17.02.19* | Решение проблемы масштабирования изображения на различных типах сканеров |  |  |
| *18.02.19* | Решение проблемы сильной разницы отработки морофологических преобразований изображения, распечатанного на разных типах принтеров |  |  |
| *19.02.19* | Коррекция угла наклона отсканированного изображения с использованием преобразования Хафа |  |  |
| *20.02.19* | Разработка алгоритмов конвертации изображения из PDF в PNG и определения таблиц |  |  |
| *21.* | Написание программы для определения таблиц в заданном контуре |  |  |
|  | Поиск и решение задачи корректного распознавания ячеек Tessarect’ом при помощи пропорционального попиксельного сдвига изображения |  |  |
|  | Попытка оптимизации алгоритма обрезки контура изображения |  |  |
|  | Перепись программы под ООП, изменение логики взаимодействия классов и их методов |  |  |
|  | Продолжение процесса переписи программы под ООП: разграничение методов класса parser\_class для каждого типа страницы, вынесение некоторых методов в статический класс util\_class |  |  |
|  | Переход к алгоритму Хафа для оптимизации времени распознавания текста |  |  |
|  | Знакомство с системами управления проектами, основными методологиями разработки проектов: Agile, PRINCE2, Scrum. Начало использования сервиса Wrike.com в качестве СУП, повторное знакомство с Redmine |  |  |
|  | Знакомство с dockstrings и системами автоматического генерирования документации, использование pydoc, документирование исходных модулей программы |  |  |
|  | Переписывание исходной документации для САГД в связи с необходимостью добавления входных и выходных параметров :param name: и :return:, пробная регенерация исходной документации, знакомство с sphinx |  |  |
|  | Анализ алгоритма формирования табличной квадратичной матрицы, обработка ошибки, при которой в один контур вписывается несколько прямоугольников |  |  |
|  | Частичная замена Tesseract OCR на Google Cloud Vision OCR в связи с большим быстродействием последнего вследствии упрощения процессинга изображения на стороне клиента, проработка механизма передачи изображения numpy array на сторону google.cloud без сохранения изображения на жесткий диск |  |  |
|  | Разграничение классов по модулям программы, общая реструктуризация и оптимизация исходного кода, вынесение общих частей кода в отдельные методы |  |  |
|  | Исправление ошибки, при которой клиент google vision делал разрыв SSL-соединения, исправление путем вынесения авторизации клиента за статический класс программы |  |  |
|  | Изучения командной оболочки для интерактивных вычислений Jupyter Notebook, описание алгоритма преобразования матрицы с использованием Python3 в JN |  |  |
|  | Разработка метода учета погрешности +- 5-10 пикселей для элементов, стоящих в одном столбце/строке, начало формирования большого ассоциативного массива с кодами строк в формах бухгалтерской отчетности |  |  |
|  | Проектирование структуры отдачи JSON, введение поля qc (quality code), вынесение логики преобразования в отдельный модуль, реструктурирование иерархии вызова модулей, общая реструктуризация в git |  |  |
|  | Разработка Supervisor’а для отслеживания поля qc как для всего документа, так и для каждого ключа последнего уровня |  |  |
|  | Добавление модуля отдачи JSON в общую программу, интеграционное тестирование взаимодействия модулей Opencv -> OKUD -> MatrixToJSON |  |  |
|  | Продолжение работы с модулем MatrixToJSON: Исправление проверки на вхождение ключей словаря MonthAndNumbers.yaml, исключение отдачи header’ов в первом элементе ключа codes, доработка фильтрации кодов формы путем поиска вхождения ключей словаря CodesAndNames.yaml. Переработка метода GetMainDocStatus SuperVisor’а: не учитываем абсолютное совпадение индексов элементов в столбце т.к. в частных случаях значение может быть таким же, как и ключ в словаре. |  |  |
|  | Вынесение логики парсинга YAML-словарей в модуль DictionaryGetter в качестве псевдостатического класса c полями, закрепление практического опыта по откату коммитов с помощью git revert, использование gitzip. Исправление ошибки неправильного определения статуса документа в OKUD. OKUD\_PROCESSING. MatrixProcessing |  |  |
|  | Тестирование программы на других PDF-документах, формирование новых задач в Wrike по оптимизации алгоритмов: обнаружение проблем с методами HoughCheck, detect\_box в scaner\_fix\_class в исполняемом модуле opencv, проблема с определением 5-и значных кодов в отдаче JSON класса ToJSON модуля MatrixToJson, а также проблема с артефактами сканирования изображения в частных случаях. |  |  |
| Результат производственной практики по модулю ПМ.03«Участие в интеграции программных модулей» | | | Оценка (зачет/незачет) | Подпись руководителя от предприятия (М.П.) |
|  |  |
| 23.03.19 – 19.04.19 |  | Разработка тестового метода для избавления от артефактов сканирования изображения, решение проблемы с разворотом изображения на 90 градусов в HoughCheck, решение проблем обрезки в detect\_box, реструктуризация текущих задач в Wrike, анализ корня проблемы в MatrixToJson |  |  |
|  | Решение проблемы с методом Counter в SuperVisor.GetMainDocStatus, вызываемый в MatrixToJson, анализ целесообразности внедрения модуля от артефактов сканирования в программу, начало разработки режима дебага с разграничением по модулям, вынесение логики подсчета Counter в метод UtilModule.util\_class.get\_max\_counter, обработка символа “-” в GetCodeQCStatus |  |  |
|  | Приватизация методов классов модуля OKUD, усовершенствование логики замены кодов: теперь используется промежуточный словарь CodeCounter с ключом по непреобразованному элементу |  |  |
|  | Разработка логики для определения продолжения страницы, если нет верхнего правого колонтитула (разработка 5 типа документа), переработка логики в MatrixToJSON: т.к у документов 5 типа нет header’ов, то берем их и предыдущего отданного JSON, тип документа которого не равен 5. |  |  |
|  | Добавление поля hindex в для формирования элементов матрицы 5 типа, добавление поля info для отдачи данных из smalltable, написание статического метода GetSmallTableQCStatus в SuperVisor для получения qc поля info, общая реструктуризация исходного кода |  |  |
|  | Обработка случая, когда smalltable не существует (документ не относится к 1 или 3 типу), вынесение логики двух типов обрезок в UtilModule и создание их новых методов: CropImager\_MinAreaRect и CropImager\_BoundingRect. Реализация механизма частичного дебага в UtilModule |  |  |
|  | Введение отдельной тестовой директории для модуля TableRecognition, перемещение метода обработки шума на изображении NoizeRemover в класс ScanerFixClass, создание отдельной директории для хранения файлов PDF. |  |  |
|  | Создание алгоритма для избавления от рамок таблиц, которые плохо влияли на распознавание Tesseract. Очередной отказ от Google Cloud Vision в пользу Tesseract, вызов NoizeRemover в цикле while для решения проблем с некорректным распознаванием таблиц в модуле TableRecognition, исправление преобразования цветового пространства изображения в NoizeRemover, попытка разработки алгоритма автодополнения ячеек матрицы на основе текущих точек и порядкового индекса элемента. |  |  |
|  | Вынесение вызова распознавания таблиц в OKUD\_PROCESSING, ввод ограничительных коэффициентов для метода NoizeRemover |  |  |
|  | Интеграция Flask и Redis в текущее решение, создание класса MainProcessingClass для вызова всей логики распознавания, ознакомление с модулем threading для управления потоками данных программы, асинхронный вызов MainProcessingClass при передаче PDF в Flask, реструктуризация логики взаимодействия Flask со скриптом распознавания, использование UUID для генерации уникального идентификатора распознаваемого документа. |  |  |
|  | Дополнительная интеграция скрипта распознавания с Redis для отдачи результатов распознавания без использования механизма синхронизации/блокировки потоков. |  |  |
|  | Формирование общего выходного JSON со всех распознанных страниц в одно целое для записи в Redis, введение поля status в качестве отдачи флага JSON до того, как поток передаст результат распознавания в СУБД. |  |  |
|  | Изучение механизма очередей в Python на базе Redis, использование класса multiprocessing.Queue, использование более быстрого файла .traineddata для Tesseract OCR |  |  |
|  | Развертка решения в docker-контейнере: добавление зависимостей, создание Dockerfile на базе Ubuntu:18.04 и docker-compose |  |  |
|  | Фикс проблемы с методом обрезки изображения: замена цвета background на белый и как следствие улучшение качества распознавания текста у Tesseract’а |  |  |
|  | Использование группировки по значениям для определения границ квадратной матрицы (метод GetGroupValues в классе UtilClass), создание временной директории файлов для каждого потока программы, замена os.remove на shutil.rmtree, удаление обращение к TABLE\_OUTPUT\_OBJ в ParserClass, разработка логики тестирования для большого количества файлов PDF |  |  |
|  | Введение метода MainMatrixStructuring в OKUD для перебора возможных пропорций квадратной матрицы, введение еще одного механизма группировки GetGroupAlterValues в UtilClass, разработка алгоритма по группировке матрицы кооринат |  |  |
|  | Разработка программы для тестирования TableRecognation и отдаваемых им значений в center\_and\_text. Реструктуризация архитектуры Docker. Начало реализация механизма очередей на базе rq-python |  |  |
|  | Продолжение переработки архитектуры Docker: перенос логики распознавания в worker, создание Dockerfile для панели управления rq-dashboard, вынесение Flask API в отдельный контейнер, линковка контейнеров между собой и создание общего docker-composer, реализация передачи файлов между web и worker с помощью файла в формате Base64, разработка логики получения результата и/или статуса распознаваемого документа с помощью job.meta |  |  |
|  |  |  |  |
| Результат производственной практики по модулю ПМ.05«Программное обеспечение компьютерных сетей» | | | Оценка (зачет/незачет) | Подпись руководителя от предприятия (М.П.) |
|  |  |

**Рекомендации и замечания руководителей практики от колледжа в период прохождения студентом практики**

**Период производственной практики:** *12.01.19 – 15.02.19*

**Профессиональный модуль:** ПМ.02 «Разработка и администрирование баз данных»

**Руководитель от колледжа:** Демкина Н.И.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Подпись руководителя

практики от колледжа \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / *Демкина Н.И.*/

**Период производственной практики:** *16.02.19 – 22.03.19*

**Профессиональный модуль:** ПМ.03 «Участие в интеграции программных модулей»

**Руководитель от колледжа:** Демкина Н.И.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Подпись руководителя**

**практики от колледжа \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /** *Демкина Н.И.***/**

**Период производственной практики:** *23.03.19 – 19.04.19*

**Профессиональный модуль:** ПМ.05 «Программное обеспечение компьютерных систем»

**Руководитель от колледжа:** Демкина Н.И.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Подпись руководителя**

**практики от колледжа \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /** *Демкина Н.И***./**

**Оценка производственной практики студента**

**Зам. директора по УПР \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/***Фокина Л.В.***/** печать колледжа