

Скоков
Роман
Юрьевич 2026.02.27
Ректор " 15:42:47
ВИЭПП" +03'00'

Приложение 1 к рабочей программе дисциплины
«Визуальное программирование на языке Python»

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОЛЖСКИЙ ИНСТИТУТ ЭКОНОМИКИ, ПЕДАГОГИКИ И ПРАВА»

Факультет информационной безопасности и права

**Методические указания для обучающихся
по освоению дисциплины
«Визуальное программирование на языке Python»**

Направление подготовки:
09.03.03 Прикладная информатика

Профиль образовательной программы:
«Прикладная информатика в юриспруденции»

Уровень высшего образования:
бакалавриат

Квалификация выпускника:
«бакалавр»

Содержание

1 Методические указания для обучающихся по планированию и организации времени, необходимого для освоения дисциплины.....	3
2 Методические указания по работе обучающихся во время проведения лекций.....	3
3 Методические указания для обучающихся по подготовке к практическим (семинарским) занятиям	3
4 Методические указания для обучающихся по организации самостоятельной работы.....	4
4.1 Методические указания по написанию реферата	4
4.2 Методические указания по написанию эссе.....	4
4.3 Методические указания по оформлению контрольной работы, эссе, реферата.....	5
4.4 Методические указания по подготовке презентации.....	5
5 Методические указания для обучающихся по организации и проведению обучения в интерактивных формах	6
5.1 Методические указания по проведению круглого стола.....	6
5.2 Методические указания по проведению лекции-дискуссии.....	6
5.3 Методические указания по проведению коллоквиума.....	7
5.4 Методические указания для проведения анализа конкретных ситуаций.....	7

1 Методические рекомендации для обучающихся по планированию и организации времени, необходимого для освоения дисциплины.

Планирование и организация времени, необходимого на изучение дисциплины «Визуальное программирование на языке Python», регламентируются учебным планом и рабочей программой. Для очной формы обучения (6 семестр) общий объем составляет 144 академических часа: лекции — 36 ч, практические занятия — 36 ч (включая 16 ч интерактивных), самостоятельная работа — 72 ч. Для заочной формы обучения (4 семестр): лекции — 4 ч, практические занятия — 8 ч (включая 4 ч интерактивных), самостоятельная работа — 132 ч.

Рекомендуется распределить время следующим образом:

После каждой лекции (2 акад. ч) выделять 1–2 часа на конспектирование и закрепление материала;

Перед практическим занятием — 1–2 часа на изучение документации библиотек и подготовку кода;

Еженедельно — 4–6 часов на выполнение домашних заданий и проектной деятельности;

Для заочников — сосредоточить основные усилия на глубоком изучении теоретических основ и самостоятельной реализации проектов в условиях ограниченного контактного времени.

Особое внимание уделить регулярной практике написания кода, работе с официальной документацией библиотек (Tkinter, PyQt6, Kivy) и использованию систем контроля версий (Git).

2 Методические указания по работе обучающихся во время проведения лекций

Дисциплина «Визуальное программирование на языке Python» формирует у обучающихся компетенции в области разработки приложений с графическим интерфейсом пользователя, что является ключевым навыком для обеспечения работоспособности программного обеспечения (ПК-2). Успешное освоение дисциплины требует систематической самостоятельной работы, особенно в части практического программирования.

Во время лекций рекомендуется:

Вести конспектирование с акцентом на архитектурные концепции (событийно-ориентированное программирование, паттерны MVC, Observer), синтаксические конструкции и особенности библиотек;

Фиксировать примеры кода, диаграммы архитектуры, схемы обработки событий;

Обращать внимание на сравнительный анализ библиотек (Tkinter vs PyQt6 vs Kivy), их преимущества и ограничения;

Задавать уточняющие вопросы по практическому применению теоретических концепций.

3 Методические указания для обучающихся по подготовке к практическим (семинарским) занятиям

Практические занятия направлены на формирование навыков разработки GUI-приложений, отладки, тестирования и проектирования архитектуры. Подготовка включает:

1. Изучение теоретического материала по соответствующей теме лекции.
2. Анализ документации библиотек (официальная документация Tkinter, PyQt6, Kivy).
3. Подготовку кода — написание заготовок решений задач, изучение примеров.
4. Подготовку вопросов по непонятным моментам для уточнения на занятии.

Планы проведения лекционных занятий

Раздел 1. Введение в визуальное программирование

Тема 1.1. Концепции GUI, архитектура событийно-ориентированных приложений
Эволюция графических интерфейсов

Цикл обработки событий (event loop)
Архитектурные паттерны: MVC, Observer в контексте GUI
Жизненный цикл GUI-приложения
Тема 1.2. Среда разработки. Настройка проекта. Структура приложения
Выбор и настройка IDE (PyCharm, VS Code)
Создание виртуального окружения
Структура проекта: модули, пакеты, ресурсы
Интеграция с системой контроля версий (Git)

Раздел 2. Библиотека Tkinter

Тема 2.1. Базовые виджеты, компоновка, обработка событий
Label, Button, Entry, Frame
Менеджеры компоновки: pack, grid, place
Привязка событий: bind, command
Тема 2.2. Диалоговые окна, меню, работа с файлами
messagebox, filedialog, Menu
Создание многооконных приложений
Обработка ошибок при работе с файлами
Тема 2.3. Многопоточность в Tkinter
Проблема блокировки интерфейса
Паттерн «Очередь событий»
Использование queue, after для асинхронных операций

Раздел 3. Библиотека PyQt6

Тема 3.1. Архитектура Qt. Signals & Slots. Qt Designer
QApplication, QMainWindow
Механизм сигналов и слотов
Визуальное проектирование интерфейсов
Тема 3.2. Сложные виджеты: таблицы, деревья, вкладки
QTableView, QTreeView, QTabWidget
Модель-представление (Model-View)
Делегаты и кастомизация отображения
Тема 3.3. Стилизация через QSS. Кастомные виджеты
Применение Qt Style Sheets
Создание пользовательских виджетов через наследование

Раздел 4. Кроссплатформенная разработка (Kivy)

Тема 4.1. KV-язык, мультитач-интерфейсы, анимации
Декларативное описание интерфейса
Свойства, события, привязки
Анимации с использованием Animation
Тема 4.2. Деплоймент приложений
Упаковка в .exe (PyInstaller)
Настройка spес-файлов
Особенности сборки для мобильных платформ

Раздел 5. Интеграция и тестирование

Тема 5.1. Подключение SQLite/PostgreSQL к GUI
Работа с базами данных через sqlite3, psycopg2
Отображение данных в таблицах
Обработка транзакций
Тема 5.2. Тестирование интерфейсов

Модульное тестирование (unittest, pytest-qt)
 Мокирование событий
 Отчеты о покрытии кода
 Тема 5.3. Профилирование и оптимизация
 Анализ производительности (cProfile)
 Оптимизация отрисовки
 Кэширование и управление ресурсами

4. Методические указания для обучающихся по организации самостоятельной работы

Самостоятельная работа включает: конспектирование лекций, подготовку к практическим занятиям, выполнение домашних заданий, написание рефератов, разработку проектов, подготовку к зачету.

4.1. Методические указания по написанию реферата

Тематика рефератов по дисциплине:

- Сравнительный анализ библиотек GUI для Python: Tkinter, PyQt, Kivy
- Паттерны проектирования в разработке GUI-приложений
- Современные подходы к тестированию графических интерфейсов
- Инструменты профилирования и оптимизации Python-приложений
- Особенности разработки кроссплатформенных приложений
- История эволюции графических интерфейсов пользователя
- Применение визуального программирования в юридической сфере
- Безопасность GUI-приложений: уязвимости и методы защиты

Структура реферата:

- Титульный лист
- Введение (актуальность, цели, задачи) — 0,5–1 стр.
- Основная часть (2–3 раздела с подзаголовками) — 3–4 стр.
- Заключение (выводы, перспективы) — 0,5–1 стр.
- Список литературы (не менее 5 источников)
- Приложения (при наличии: фрагменты кода, скриншоты интерфейсов)

Критерии оценки:

- Соответствие теме и плану
- Глубина анализа источников
- Логичность изложения
- Наличие собственных выводов
- Грамотное оформление

4.2. Методические указания по написанию эссе

Темы для эссе:

- «Будущее визуального программирования: от десктопа к метавселенным»
- «Как GUI влияет на пользовательский опыт в юридических приложениях?»
- «Этика проектирования интерфейсов: баланс между функциональностью и доступностью»
- «Мой путь в GUI-разработке: ошибки, открытия, достижения»

Структура эссе:

- Титульный лист (нумерация начинается с этой страницы, номер не ставится)
- Введение (постановка проблемы, актуальность) — 1 стр.
- Основная часть (личные размышления, аргументы, примеры из практики) — 3–4 стр.
- Заключение (выводы, рекомендации) — 1 стр.
- Список литературы (не менее 3 источников)

Эссе должно отражать индивидуальную позицию автора, подкрепленную аргументами и примерами из учебной или проектной деятельности.

4.3. Методические указания по оформлению контрольной работы, эссе, реферата

Требования к оформлению:

- Формат: A4, поля: верхнее/нижнее — 2 см, левое — 3 см, правое — 1,5 см
- Шрифт: Times New Roman, 14 pt
- Межстрочный интервал: 1,5
- Абзацный отступ: 1,25 см
- Выравнивание: по ширине
- Нумерация страниц: сквозная, внизу по центру, начиная со второй страницы
- Ссылки на источники — постраничные сноски или в квадратных скобках
- Код в приложениях — моноширинный шрифт (Courier New, 10 pt) с подсветкой синтаксиса

Критерии оценки:

- «Отлично»: полное соответствие теме, глубокий анализ, грамотное оформление, оригинальные выводы
- «Хорошо»: соответствие теме, достаточный анализ, незначительные замечания по оформлению
- «Удовлетворительно»: частичное раскрытие темы, поверхностный анализ, замечания по содержанию и оформлению
- «Неудовлетворительно»: несоответствие теме, отсутствие анализа, грубые ошибки в оформлении

4.4. Методические указания по подготовке презентации

Структура презентации проекта:

1. Титульный слайд (название проекта, ФИО автора, группа)
2. Цель и задачи проекта
3. Обзор аналогов и обоснование выбора технологий
4. Архитектура приложения (диаграмма)
5. Демонстрация ключевых функций (скриншоты/видео)
6. Особенности реализации (паттерны, алгоритмы)
7. Результаты тестирования
8. Выводы и перспективы развития

Технические требования:

- Шрифт: заголовки — не менее 28 pt, основной текст — не менее 24 pt
- Цветовая схема: контрастная, без излишней пестроты
- Количество слайдов: 8–12
- Использование анимации — умеренно, только для акцента
- Все изображения — высокого качества, с подписями

5. Методические указания для обучающихся по организации и проведению обучения в интерактивных формах

5.1. Методические указания по проведению круглого стола

Тема для обсуждения: «Выбор библиотеки GUI для корпоративного приложения юридической фирмы: критерии и компромиссы»

Цель: Выработать коллективные рекомендации по выбору технологического стека с учетом требований: кроссплатформенность, безопасность, поддержка тем оформления, интеграция с базами данных.

Этапы проведения:

1. Подготовительный: распределение ролей (представители команд, отстаивающих Tkinter, PyQt6, Kivy), подготовка аргументов.
2. Основной: презентация позиций, дискуссия, поиск компромиссов.
3. Заключительный: формулирование итоговых рекомендаций, голосование.

Критерии оценки:

- Активность участия
- Обоснованность аргументов
- Умение вести дискуссию и уважать оппонентов
- Вклад в выработку итогового решения

5.2. Методические указания по проведению лекции-дискуссии

Тема: «Стилизация интерфейсов: встроенные средства vs CSS-подобные подходы (QSS)»

Цель: Сформировать критическое мышление при выборе методов кастомизации интерфейсов; развить навыки аргументации профессиональной позиции.

Методика проведения:

1. Назначение секретаря для фиксации ключевых аргументов.
2. Предварительное голосование по позициям.
3. Свободное обсуждение с обязательным обоснованием каждой позиции примерами из практики.
4. Промежуточное резюмирование преподавателем.
5. Повторное голосование и анализ изменений.
6. Формулирование практических рекомендаций.

Критерии оценки:

- Секретарь — 0,5–1,5 балла
- Активное участие с аргументированными выступлениями — 0,5–1 балл за выступление

5.3. Методические указания по проведению коллоквиума

Тема: «Архитектурные решения в разработке GUI-приложений»

Цель: Проверить глубину понимания принципов проектирования, умение обосновывать выбор архитектурных решений.

Формат: Устный опрос по билетам, включающим теоретические вопросы и практические кейсы (например: «Как вы спроектируете приложение для обработки 10 000 записей в таблице без блокировки интерфейса?»).

Критерии оценки:

- «Отлично»: всесторонняя осведомленность, четкие аргументы, примеры из практики
- «Хорошо»: хорошее знание материала, но с отдельными пробелами
- «Удовлетворительно»: базовое понимание, затруднения в аргументации
- «Неудовлетворительно»: поверхностные знания, неспособность применить теорию на практике

5.4. Методические указания для проведения анализа конкретных ситуаций

Кейс: «Приложение для юридической фирмы работает корректно на ПК разработчика, но при установке на компьютер клиента возникают ошибки отображения интерфейса и падения приложения»

Задачи для анализа:

1. Какие возможные причины проблемы?
2. Какие шаги необходимо предпринять для диагностики?
3. Какие меры предотвратят подобные проблемы в будущем?
4. Как организовать процесс тестирования на разных конфигурациях?

Этапы работы:

1. Индивидуальный анализ ситуации (10 мин)
2. Обсуждение в микрогруппах (15 мин)
3. Представление решений от групп (15 мин)
4. Коллективное обсуждение и формулирование итоговых рекомендаций (10 мин)

Критерии оценки:

- Глубина анализа проблемы
- Практическая применимость предложенных решений
- Умение работать в команде
- Качество презентации решения