

Министерство науки и высшего образования Российской Федерации

**Федеральное государственное автономное образовательное учреждение
высшего образования**

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**

Факультет дистанционного обучения (ФДО)

Р. С. Кульшин

ИНТЕРФЕЙСЫ ПРОГРАММНЫХ СИСТЕМ

**Методические указания
по организации самостоятельной работы
и выполнению лабораторной работы для обучающихся
с применением дистанционных образовательных технологий**

Томск 2025

Корректор: А. Н. Миронова

Кульшин Р. С.

Интерфейсы программных систем : методические указания по организации самостоятельной работы и выполнению лабораторной работы для обучающихся с применением дистанционных образовательных технологий / Р. С. Кульшин. – Томск : ФДО, ТУСУР, 2025. – 28 с.

© Кульшин Р. С., 2025

© Оформление.

ФДО, ТУСУР, 2025

ОГЛАВЛЕНИЕ

Введение	4
1 Организация самостоятельной работы	6
1.1 Общие положения	6
1.2 Рабочая программа и план обучения	8
1.3 Теоретический материал	8
1.4 Рекомендации по работе с учебной и научной литературой	9
1.5 Консультации	13
1.6 Лабораторные работы	13
1.7 Контрольные мероприятия	15
1.8 Итоговая аттестация	15
2 Методические указания по выполнению лабораторной работы	
«Проектирование интерфейса программной системы»	16
2.1 Теоретическая часть	16
2.2 Порядок выполнения лабораторной работы	20
2.2.1 Установка ПО	20
2.2.2 Задание на лабораторную работу	21
2.2.3 Варианты заданий на лабораторную работу	24
2.2.4 Вопросы для самоконтроля	24
2.3 Требования к содержанию и оформлению отчета	25
Литература	27
Приложение А Пример оформления титульного листа отчета	28

ВВЕДЕНИЕ

В современном цифровом обществе удобство и эффективность взаимодействия человека с программными системами становятся одним из ключевых факторов успеха программного продукта. Интерфейс программной системы – это та часть программы, с которой непосредственно сталкивается пользователь. Именно от качества проектирования интерфейса во многом зависит впечатление пользователя, его продуктивность, а также востребованность самой системы. Под интерфейсом программной системы понимается не только набор графических элементов (окна, кнопки, меню), но и вся система взаимодействия: навигация, структура информации, визуальная иерархия, отклик на действия пользователя, доступность, адаптивность под разные устройства.

Современные программные системы ориентированы на создание интуитивно понятных, эстетически привлекательных и универсальных интерфейсов, соответствующих принципам user experience (UX) и user interface design (UI). Проектирование интерфейсов программных систем основано на использовании различных методологий и стандартов, таких как Material Design, Human Interface Guidelines, а также принципах эргономики и когнитивной психологии. Ключевыми этапами процесса разработки интерфейсов являются анализ потребностей пользователя, создание прототипов и макетов, проведение юзабилити-тестирования и постоянная итерация дизайна на основе обратной связи.

В процессе проектирования интерфейсов используются современные инструменты и среды (Figma, Adobe XD, Sketch), а также фреймворки и библиотеки, обеспечивающие реализацию дизайна в реальных программных продуктах (например, Flutter, React, WPF, Qt). Большое внимание уделяется вопросам доступности, адаптивности интерфейса под различные платформы и устройства, а также визуальной согласованности и брендированию.

Важная часть дисциплины посвящена не только визуальной стороне, но и вопросам проектирования сценариев взаимодействия, созданию эффективной структуры пользовательских потоков, проработке логики интерфейса, согласованности элементов управления и унификации пользовательских паттернов. Современные подходы к дизайну интерфейсов программных систем ориентированы на максимальное упрощение взаимодействия, снижение когнитивной нагрузки, обеспечение предсказуемости и удобства работы с программой. Для этого используются принципы модульности, повторного использования компонентов, системного мышления и многократного тестирования прототипов с реальными пользователями.

Таким образом, дисциплина «Интерфейсы программных систем» формирует у студентов знания и навыки в области проектирования современных пользовательских интерфейсов, а также понимание важности UX/UI в создании конкурентоспособных программных продуктов. Освоение этой дисциплины позволяет будущим разработчикам и дизайнерам создавать удобные, функциональные и эстетичные интерфейсы, отвечающие ожиданиям пользователей и требованиям рынка.

Цели дисциплины «Интерфейсы программных систем»:

- ознакомление с современными принципами и инструментами дизайна пользовательских интерфейсов;
- формирование практических навыков проектирования, прототипирования и оценки интерфейсов программных систем;
- получение практических навыков проектирования и реализации интерфейсов для различных типов программных систем в ходе выполнения лабораторных работ.

1 ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

1.1 Общие положения

Самостоятельная работа – основной вид учебной деятельности студентов ФДО ТУСУР, который позволяет овладевать фундаментальными знаниями, профессиональными умениями и навыками в конкретных областях науки и (или) виде (видах) деятельности, на которые ориентирована программа бакалавриата; получать опыт творческой и исследовательской деятельности; развивать такие качества личности, как самостоятельность в принятии решений, ответственность, организованность; использовать творческий подход к решению проблем учебного и профессионального уровней.

Задачи самостоятельной работы студента:

- систематизация и закрепление полученных теоретических знаний и практических умений;
- углубление и расширение теоретической подготовки; формирование умения использовать нормативную, правовую, справочную документацию и специальную литературу;
- развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений;
- использование материала, собранного и полученного в ходе самостоятельных занятий, при написании курсовых и выпускной квалификационной работ, для эффективной подготовки к зачетам и экзаменам.

Основные черты самостоятельной деятельности в высшей школе:

- *целенаправленность*, т. е. такая активность, которая направлена на достижение сознательно поставленной цели;

- *продуманность*, т. е. анализ ситуации и выбор способов и средств достижения цели и последовательности будущих действий;
- *осознанность*, планирование и предвидение возможного результата, наличие логической схемы;
- *структурность*, специфический набор действий и последовательность их осуществления;
- *результативность*, когда деятельность находит свое завершение в результате.

Целью предлагаемых методических указаний является знакомство с порядком организации самостоятельной работы по изучению дисциплины с применением дистанционных образовательных технологий.

В рабочем учебном плане (<https://edu.tusur.ru/opops>) приведен перечень компетенций, формируемых в результате изучения дисциплины, а также указана совокупность тех знаний, умений и навыков, которые должны стать результатом освоения материалов курса.

Изучение дисциплины «Интерфейсы программных систем» включает следующие виды деятельности:

- 1) знакомство с рабочей программой дисциплины;
- 2) составление плана обучения по дисциплине;
- 3) изучение теоретического материала;
- 4) выполнение лабораторных работ;
- 5) прохождение промежуточной аттестации (контрольная работа);
- 6) прохождение итоговой аттестации (зачет).

Необходимо отметить, что приведенный перечень не предполагает строгой последовательности действий. На любом этапе возможно внесение корректировок в план работы. Возвращение к теоретическому материалу необходимо при подготовке ко всем видам контрольных мероприятий, итоговой аттестации и т. д.

Все теоретические и вспомогательные материалы по дисциплине размещаются в соответствующем электронном курсе.

1.2 Рабочая программа и план обучения

Рабочая программа дисциплины – программа освоения учебного материала, соответствующая требованиям государственного образовательного стандарта высшего образования и учитывающая специфику подготовки студентов по выбранному направлению.

В рабочей программе дана общая информация по дисциплине: цели и задачи дисциплины, место в образовательной программе, требование к результатам освоения дисциплины (на формирование каких компетенций направлена), объем и виды учебной работы, формы аттестации, основная и дополнительная литература, примерные оценочные материалы.

Рабочая программа позволяет представить объем предстоящей работы по дисциплине и составить план самостоятельного обучения. План должен включать следующие важные составляющие: 1) изучение теоретического материала дисциплины, освоение терминологии, ознакомление с классификациями и др. (для этого необходимо оценить объем материалов электронного курса и дополнительной литературы); 2) подготовка и выполнение контрольных работ; 3) подготовка и прохождение итоговой аттестации по дисциплине.

Рабочая программа дисциплины доступна из рабочего учебного плана соответствующей основной профессиональной образовательной программы (<https://edu.tusur.ru/opops>).

1.3 Теоретический материал

Теоретический материал представлен в различных форматах:

- тексто-графические материалы;
- видеоматериалы;
- презентации.

Тексто-графические материалы

Теоретический материал по дисциплине «Интерфейсы программных систем» размещен в электронном курсе.

Самостоятельная работа предполагает обязательное изучение рекомендованной литературы, выполнение практических заданий и закрепление усвоенного материала. Для углубленного изучения тем рекомендуется использовать дополнительные литературные источники.

Для удобства обучающихся предоставляется доступ из личного кабинета в [электронную библиотеку ФДО](#), в которой представлены все разработанные на факультете учебные материалы.

Студенты ФДО имеют доступ к полным текстам изданий, размещенных на образовательной платформе «[Юрайт](#)» и в электронной библиотечной системе «[Лань](#)».

Презентации

Некоторые темы электронных курсов снабжены дополнительным материалом, представленным в виде презентаций. Этот вид методического обеспечения дисциплины позволяет наглядно представить обобщенную информацию по теме, практический разбор задач или ознакомить с новым материалом, требующим визуализации.

Для получения разностороннего представления о содержании дисциплины рекомендуется изучить все виды теоретического материала.

1.4 Рекомендации по работе с учебной и научной литературой

Самостоятельная работа по освоению теоретического материала учебного пособия и другой учебной и научной литературы (а также самостоятельное теоретическое исследование изучаемых проблем) – это важнейшее условие

формирования научного способа познания. Таким образом, чтение научного текста является частью познавательной деятельности, главная цель которой – извлечение из текста необходимой информации. Осознание читающим внутренней установки при обращении к печатному слову (выделение нужных сведений, усвоение информации полностью или частично, критический анализ материала и т. п.) определяет эффективность осуществляемого действия.

Грамотная работа с книгой, особенно если речь идет о научной литературе, предполагает *соблюдение ряда правил*, для овладения которыми необходимо учиться. Прежде всего, при такой работе невозможен формальный, поверхностный подход. Главное правило при работе над книгой – не механическое заучивание, не простое накопление цитат, выдержек, а сознательное усвоение прочитанного, осмысление его, стремление дойти до сути.

Изучение книги должно происходить в определенной последовательности. Вначале следует ознакомиться с оглавлением (содержанием), предисловием, введением. Это позволяет получить общее представление о структуре и вопросах, которые рассматриваются в книге. Затем можно переходить непосредственно к чтению. Чтобы получить цельное представление о книге, первый раз стоит прочитать ее от начала до конца. При повторном чтении происходит постепенное глубокое осмысление каждой главы, критического материала и позитивного изложения; выделение основных идей, системы аргументов, наиболее ярких примеров и т. д. Непременным правилом чтения должно быть выяснение незнакомых слов, терминов, выражений, неизвестных имен, названий. Важная роль в связи с этим принадлежит библиографической подготовке студентов. Она включает в себя умение активно, быстро пользоваться научным аппаратом книги, справочными изданиями, каталогами; вести поиск необходимой информации, обрабатывать и систематизировать ее.

Рекомендуется использовать четыре основные установки в чтении научного текста:

- информационно-поисковая (задача – найти, выделить искомую информацию);
- усваивающая (усилия читателя направлены на то, чтобы как можно полнее осознать и запомнить как сами сведения, излагаемые автором, так и всю логику его рассуждений);
- аналитико-критическая (читатель стремится критически осмыслить материал, проанализировав его, определив свое отношение к нему);
- творческая (создает у читателя готовность в том или ином виде – как отправной пункт для своих рассуждений, как образ для действия по аналогии и т. п. – использовать суждения автора, ход его мыслей, результат наблюдения, разработанную методику, дополнить их, подвергнуть новой проверке).

Научная методика работы с литературой предусматривает фиксирование прочитанной информации в систематизированных записях разного рода. Это позволяет привести в систему знания, полученные при чтении, сосредоточить внимание на главных положениях, зафиксировать, закрепить их в памяти, а при необходимости вновь обратиться к ним.

Основные виды систематизированной записи прочитанного:

- тезирование – лаконичное воспроизведение основных утверждений автора без привлечения фактического материала;
- цитирование – дословное выписывание из текста выдержек, извлечений, наиболее существенно отражающих ту или иную мысль автора;
- конспектирование – краткое и последовательное изложение содержания прочитанного.

Методические рекомендации по составлению конспекта

Внимательно прочитайте текст. Уточните в глоссарии или справочной литературе непонятные слова. Выделите главное, составьте план, представляющий собой перечень заголовков, подзаголовков, вопросов, последовательно раскрываемых затем в конспекте. Это первый элемент конспекта.

Вторым элементом конспекта являются тезисы. Тезис – это кратко сформулированное положение. Для лучшего усвоения и запоминания материала следует записывать тезисы своими словами. Тезисы, выдвигаемые в конспекте, нужно доказывать.

Третий элемент конспекта – основные доводы, доказывающие истинность рассматриваемого тезиса. В конспекте могут быть положения и примеры.

Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли.

При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного.

Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнений и дополнений необходимо оставлять поля.

Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы. Конспект ускоряет повторение материала, экономит время при повторном обращении к работе.

1.5 Консультации

В связи с тем что не всегда возможно самостоятельно найти ответы на возникшие в ходе изучения дисциплины вопросы, на факультете существует возможность задать вопрос преподавателю, записавшись к нему на консультацию. Вопросы можно задавать как по теоретическому материалу, так и по подготовке, оформлению отчетов любого вида работ.

Предусмотрено несколько ресурсов, где можно задать вопрос:

- на учебном форуме, воспользовавшись формой отправки из учебного плана или почтового агента в личном кабинете;
- написав письмо с любого другого почтового сервера на адрес problems@fdo.tusur.ru.

Также существует возможность обсудить любой учебный вопрос на форуме с другими студентами, коллективно найти решение. Однако следует помнить, что мнения других студентов могут быть ошибочными, и критически относиться к высказанным идеям.

1.6 Лабораторные работы

Лабораторные работы являются основным видом деятельности, направленным на экспериментальное подтверждение теоретических положений и формирование учебных и профессиональных практических умений.

Выполнение студентами лабораторных работ необходимо:

- для формирования практических умений в соответствии с требованиями к уровню подготовки студентов, установленными рабочей программой дисциплины по конкретным разделам (темам);
- обобщения, систематизации, углубления, закрепления полученных теоретических знаний;

- совершенствования умений применять полученные знания на практике, реализации единства интеллектуальной и практической деятельности;
- развития интеллектуальных умений;
- выработки при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

Лабораторная работа

Лабораторные работы позволяют объединить теоретико-методологические знания и практические навыки учащихся в процессе научно-исследовательской деятельности.

По дисциплине «Интерфейсы программных систем» предусмотрено выполнение текстовой лабораторной работы.

Задание к текстовой лабораторной работе располагается в учебных материалах. Для выполнения работы необходимо повторить уже усвоенные разделы учебного пособия по теме, изучить дополнительный материал, который может быть дан в методических указаниях и/или дополнительной литературе. Необходимо:

- подробно изучить текст задания и определить свой вариант;
- обязательно оформить отчет в соответствии с представленными требованиями;
- отправить его на проверку преподавателю.

Для выполнения лабораторной работы требуется установка дополнительного программного обеспечения, которое можно скачать в свободном доступе.

Лабораторные работы оцениваются, получение зачета является обязательным условием для аттестации по дисциплине.

1.7 Контрольные мероприятия

Контрольная работа

Цель выполнения контрольной работы – закрепить знания, полученные студентами при изучении материалов электронного курса, а также, в ряде случаев, отработать навыки решения практических задач. К выполнению контрольной работы следует серьезно подготовиться, повторив весь теоретический материал и потренировавшись в решении задач.

Дистанционное обучение ориентировано на повышенные требования к системе контроля, который при этом имеет свою специфику.

По дисциплине «Интерфейсы программных систем» предусмотрено выполнение компьютерной контрольной работы.

Компьютерная контрольная работа представляет собой совокупность тестовых заданий. Здесь могут быть представлены как теоретические вопросы, так и практические задачи. Тестовые вопросы могут быть разных типов: как на выбор одного либо нескольких вариантов, так и на ввод ответа.

1.8 Итоговая аттестация

Зачет

Для получения зачета достаточно успешно выполнить все оцениваемые работы по дисциплине (лабораторную и контрольную работы).

2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ «ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА ПРОГРАММНОЙ СИСТЕМЫ»

Целью данной лабораторной работы является освоение основных этапов проектирования пользовательского интерфейса программной системы – от построения структуры приложения до создания кликабельного прототипа.

Выполнение лабораторной работы включает в себя следующие этапы:

- определение задач работы;
- определение порядка работы или отдельных ее этапов;
- непосредственное выполнение работы студентами;
- подведение итогов и формулирование основных выводов;
- оформление отчета.

2.1 Теоретическая часть

Карта системы – это графическая схема, отражающая структуру программной системы, все основные разделы и связи между ними. В контексте проектирования интерфейсов карта системы играет роль своеобразного «плана навигации», который позволяет понять, как устроено приложение или сайт, какие экраны в нем присутствуют и как пользователь перемещается между ними.

Создание карты системы является важнейшим шагом в проектировании, поскольку именно на этом этапе фиксируются все ключевые элементы интерфейса, их иерархия и маршруты переходов. Карта помогает формализовать архитектуру приложения, заранее продумать логику пользовательских сценариев и обеспечить последовательность работы всей команды – от аналитика и дизайнера до разработчика. Она делает структуру системы наглядной и понятной, снижая риск логических ошибок на дальнейших этапах проектирования.

Как правило, карта системы оформляется в виде блок-схемы или *mind map*, где каждый элемент представляет отдельный экран или раздел приложения, а линии или стрелки показывают возможные переходы. В зависимости от задач проектирования карта может быть более или менее детализированной: иногда достаточно указать только основные разделы, а иногда важно расписать даже вспомогательные и служебные экраны.

Карта системы позволяет увидеть картину приложения целиком, выявить слабые места в логике навигации и заранее предусмотреть все возможные пути пользователя. Это особенно важно при создании сложных продуктов, где существует множество ролей и пользовательских сценариев. На базе такой карты впоследствии разрабатываются варфреймы, визуальные макеты и интерактивные прототипы.

Figma является кросс-платформенным онлайн-сервисом для реализации дизайн-прототипов и дизайн-макетов в виде онлайн-проектов, что значительно облегчает процесс передачи макетов системы в разработку. Использование пакета Figma предполагает оформление ежемесячной подписки в случае его использования в командах, состоящих из более чем двух дизайнеров. В рамках данной лабораторной работы будет достаточно бесплатного функционала.

Основными плюсами продукта Figma являются:

- работа с проектами как в онлайн- (веб-браузер), так и в десктоп-варианте;
- возможность совместной работы команды над макетами в real-time режиме;
- отсутствие необходимости передачи файлов проектов, поскольку проекты, созданные в Figma, хранятся в облаке и, следовательно, для передачи доступа к проекту с макетами достаточно лишь передать ссылку на проект. В комбинации с тем, что Figma является онлайн-приложением, действия, совершаемые над макетом, отображаются в режиме реального

времени, что устраняет необходимость актуализировать ссылку на проект после каждой итерации дизайн-процесса.

Помимо этого, в Figma присутствует большое количество функционала, облегчающего процесс создания прототипов и дизайн-макетов.

Компонент в Figma – это собранный элемент интерфейса, копии которого при изменении компонента-родителя (*main component*) он передаст всем своим копиям, или, по-другому, экземплярам (*instances*).

При этом важно помнить, что непосредственные изменения экземпляра компонента имеют более высокий приоритет по сравнению с унаследованными (например, если у экземпляра компонента кнопки изменить текст внутри, то изменение текста в родителе никак не повлияет на ранее измененный текст экземпляра, но при этом при изменении цвета кнопки родителя у компонента он также изменится, поскольку ранее экземпляр данный параметр не менял). Основное преимущество использования компонентов Figma заключается в экономии времени, т. к. дизайн интерфейса собирается из компонентов. При необходимости изменить тот или иной элемент в рамках проекта гораздо легче модифицировать родительский компонент, нежели вносить изменения во все необходимые элементы вручную, рискуя ошибиться и тем самым ввести команду проекта в замешательство из-за несоответствия элементов в рамках макета.

Помимо прочего, в Figma можно объединить компоненты в группы, создав таким образом новые варианты компонентов на основе существующих.

Варианты компонентов в Figma – это группы компонентов, связанных между собой в рамках одного элемента. Чаще всего они используются для отрисовки нескольких состояний того или иного компонента, дальнейшей передачи в разработку либо же для использования в макетах системы. Например, если нам необходимо отобразить состояние наведения на кнопку

мышкой, мы можем создать вариант данной кнопки и оформить его под необходимое состояние. При этом все варианты хранятся в группе, что структурирует расположение компонентов и повышает читаемость макетов.

Wireframe (варфрейм) – макет будущей системы, очищенный от визуальных данных, отражающий основную идею и смысл. Это минималистичное статичное отображение экранов без какой-либо визуальной проработки.

Варфрейм призван показать структуру контента, взаимодействие и расположение основных элементов навигации и границы проекта. Данный этап дает возможность представить структуру будущего проекта, не затрачивая при этом больших ресурсов на ее реализацию. Используя варфреймы, легче увидеть и исправить ошибки проектирования дизайна и логики приложения в целом.

Как было сказано ранее, варфреймы по своей сути «очищены от визуальных данных». Поэтому при их реализации применяются максимально простые фигуры и однотонные цвета, которые используются исключительно для обозначения границ элементов интерфейса либо же для выделения объектов взаимодействия с интерфейсом.

При реализации необходимо учитывать все детали системы, выделенные на прошлом этапе проектирования, поскольку если при переходе на следующий этап реализации дизайна системы в варфреймах останутся нерешенные проблемы, то цена их исправления в уже готовом дизайн-макете будет гораздо выше.

При отрисовке варфреймов экранных форм в Figma как основу необходимо использовать инструмент Frame (горячая клавиша F), при его выборе справа появятся подсказки с размерами широко распространенных устройств, по нажатию на один из которых автоматически будет создан фрейм необходимого размера.

Дизайн-макет представляет собой визуальный образ системы, отображающий структуру, расположение и характеристики отдельных элементов, что позволяет разработчикам наглядно изучить дизайн будущей системы.

Дизайн-макет предоставляет информацию о следующих элементах системы:

- параметры и размеры отдельных экранов системы;
- цветовое оформление отдельных элементов и всей системы в целом;
- количество и расположение контента на странице;
- структура и оформление отдельных функциональных элементов;
- отступы и границы элементов внутри страниц системы.

Прототип – это быстрая и упрощенная реализация будущей системы. Упрощение может быть произведено в отображаемых на макетах данных, в оформлении макета системы в целом; реализовано через удаление функций, имеющих незначительный вес в рамках реализуемого продукта. По сути, прототип – это работающая модель, либо, другими словами, опытный образец, который может тестироваться на потенциальных пользователях.

Чаще всего прототипы готовят на этапе уже реализованного дизайн-макета, что помогает увидеть функционально «урезанную» версию системы, которую можно представить заказчику для согласования либо использовать внутри команды для демонстрации логики переходов между экранами или же для понимания общего поведения системы.

2.2 Порядок выполнения лабораторной работы

2.2.1 Установка ПО

Для начала работы в Figma откройте официальный сайт <https://www.figma.com/> в браузере. В правом верхнем углу вы увидите кнопку регистрации (*Sign up* или *Зарегистрироваться*). Для создания аккаунта вы можете воспользоваться своим e-mail, Google-аккаунтом или Apple ID. Если

выбираете e-mail, укажите адрес и придумайте пароль, затем подтвердите регистрацию по ссылке из письма, которое придет на указанную почту. Если используете Google или Apple, вход произойдет через их стандартную форму. После регистрации Figma может предложить заполнить несколько дополнительных полей: имя, роль (например, студент), цель использования. Эти данные можно ввести по желанию. Затем вы сразу попадете в свой личный кабинет Figma и сможете приступить к работе.

Оказавшись на главном экране, вы увидите рабочую область с вашими черновиками и проектами. Для создания нового файла выберите на панели слева пункт *Drafts* (Черновики). Это ваше личное пространство для проектов. В верхней части экрана находится кнопка *New design file* (или значок «+»). Нажмите ее, чтобы открыть чистый холст. Сразу стоит дать файлу осмысленное название: кликните по стандартному названию *Untitled* в верхней части экрана и введите нужное имя (например: «Лабораторная работа 1»).

Перед вами откроется белый рабочий холст, где будут располагаться все элементы интерфейса. В верхней части окна находятся основные инструменты: создание фреймов, фигур, текстовых полей и прочего. Слева отображается структура вашего проекта (список экранов, групп, компонентов), справа – параметры выбранного объекта (цвет, размер, шрифты и т. д.). Вы можете сразу приступить к созданию первого экрана, используя инструмент *Frame* (клавиша *F*) или другие доступные инструменты. В Figma все сохраняется автоматически, поэтому беспокоиться о потере данных не нужно. Вернуться к списку своих файлов можно в любой момент, кликнув на логотип Figma или используя главное меню.

2.2.2 Задание на лабораторную работу

Задание состоит из четырех пунктов, каждый из которых обязателен к выполнению. Задание выполняется исходя из варианта (пп. 2.2.3). Номер варианта указан в личном кабинете студента.

1. Построение карты системы

Определить основные разделы, модули и взаимосвязи между ними, сформировать структуру программной системы.

Этапы выполнения:

1. Проанализировать требования к системе и определить ключевые сценарии использования.

2. Определить основные разделы и функциональные модули (например: главный экран, профиль пользователя, настройки, история и т. д.).

3. Нарисовать карту (mind map или блок-схему) структуры приложения с указанием связей между экранами/разделами.

4. Обосновать логику переходов между разделами.

Результат: графическая схема (карта) структуры интерфейса программной системы.

2. Создание варфрейма (wireframe)

Разработать схематичное представление (скелет) основных экранов системы без использования цвета и графических деталей в системе Figma. Требуется разработать не менее пяти варфреймов.

Этапы выполнения:

1. Для каждого ключевого экрана (определенного на этапе карты) создать варфрейм – черновую компоновку основных элементов интерфейса.

2. Обозначить расположение навигационных элементов, кнопок, полей ввода, таблиц и т. д.

3. Оформить варфреймы в едином стиле, чтобы было понятно, как пользователь будет взаимодействовать с системой.

Результат: набор варфреймов (скелетов) всех основных экранов системы.

3. Создание цветного макета (дизайн-макет)

Оформить экраны в цвете, добавить визуальные детали, отразить будущий стиль приложения.

Этапы выполнения:

1. На основе варфреймов разработать цветные макеты экранов, применяя выбранную цветовую палитру и элементы фирменного стиля.
2. Оформить все визуальные элементы (кнопки, иконки, меню, заголовки и пр.) согласно принципам современного UI-дизайна.
3. Убедиться, что цветовая схема обеспечивает читаемость и визуальный комфорт.

Результат: полноцветные макеты всех основных экранов.

4. Прототипирование (создание кликабельного прототипа)

Собрать кликабельный прототип, позволяющий имитировать работу приложения и демонстрировать пользовательские сценарии.

Этапы выполнения:

1. Настроить интерактивные переходы между экранами (нажатие на кнопки, переход по меню и т. д.).
2. Протестировать сценарии пользовательского взаимодействия в прототипе.

Результат: интерактивный (кликабельный) прототип, отражающий основные пользовательские сценарии.

Каждый этап фиксируется в виде артефактов (карта системы, варфреймы, цветные макеты, кликабельный прототип), которые предоставляются в отчете по лабораторной работе. В отчете также описываются принятые решения, использованные инструменты и особенности реализации.

2.2.3 Варианты заданий на лабораторную работу

Задание по вариантам предполагает создание интерфейса программной системы для своего проекта. Необходимо изучить предметную область и на основе полученных данных сформировать карту системы. Далее требуется выделить 5 ключевых, связанных между собой страниц системы и создать варфреймы. Затем на основе варфреймов создать цветной макет и сделать кликабельный прототип.

Варианты проектов

1. Сайт для салона красоты.
2. Сайт для больницы.
3. Сайт для библиотеки.
4. Сайт для кинотеатра.
5. Интернет-магазин спортивной одежды.
6. Новостной сайт.
7. Интернет-магазин «Аптека».
8. Сайт для проката автомобилей.
9. Интернет-магазин рыболовных товаров.
10. Сайт для школы.
11. Сайт для заказа еды.
12. Интернет-магазин канцелярских товаров.
13. Интернет-магазин мебели.
14. Сайт «Доска объявлений».
15. Интернет-магазин «Зоомагазин».

2.2.4 Вопросы для самоконтроля

1. Что такое карта системы и для чего она строится на первом этапе проектирования интерфейса?
2. Чем отличается варфрейм от цветного макета интерфейса?

3. Какие задачи решаются при создании варфрейма?
4. Почему важно уделять внимание прототипированию и тестированию пользовательских сценариев?
5. Какие инструменты и функции Figma используются для создания кликабельных прототипов?
6. В чем преимущества работы с кликабельным прототипом по сравнению с обычными статичными макетами?
7. Что такое компоненты в Figma и как они помогают при проектировании интерфейса?
8. Какие основные принципы нужно учитывать при выборе цветовой схемы для интерфейса?
9. Для чего используется и как организуется структура страниц (*Pages*) и слоев (*Layers*) в Figma?
10. Почему важно соблюдать согласованность и единый стиль элементов интерфейса во всем проекте?

2.3 Требования к содержанию и оформлению отчета

После выполнения работы студент должен представить отчет о проделанной работе с описанием полученных результатов и выводов. Отчет по лабораторным работам должен содержать следующие структурные части:

- титульный лист (1 стр.) (см. приложение А);
- содержание (1 стр.);
- цели и задачи лабораторной работы (1 стр.);
- формулировка индивидуального задания;
- раздел «Карта системы» (скриншоты работы и сопроводительные комментарии);
- раздел «Варфрейм» (скриншоты работы и сопроводительные комментарии);

- раздел «Дизайн-макет» (скриншоты работы и сопроводительные комментарии);
- раздел «Прототип» (скриншоты работы и сопроводительные комментарии);
- анализ полученных результатов;
- выводы о проделанной работе;
- приложения (при наличии).

Оформление отчета должно соответствовать требованиям Образовательного стандарта ТУСУР 01-2021 «Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления»

(<https://lkstudy.tusur.ru/book?id=293FB9187BC65D3D47258C5B001B3A8A>).

Изложение текста работы должно быть последовательным, логичным, конкретным. Необходимо привести ссылки на использованные материалы (при наличии) с указанием номера источника в списке литературы.

ЛИТЕРАТУРА

Основная литература

1. Основы проектирования интерфейсов информационных систем : электронный курс. – Томск : ФДО, ТУСУР, 2025.

Дополнительные источники

1. Окунев, А. Руководство по Figma / А. Окунев. – URL: <https://slashdesigner.ru/figma-guide> (дата обращения: 27.05.2025).

2. Спицина, И. А. Применение системного анализа при разработке пользовательского интерфейса информационных систем : учеб. пособие / И. А. Спицина, К. А. Аксенов. – Екатеринбург : Изд-во Урал. ун-та, 2018. – 100 с.

Дополнительные источники по организации самостоятельного обучения

1. Гид студента : электронный курс. – URL: <https://study.tusur.ru/tdo> (Меню студента – Гид студента).

2. Современные образовательные технологии / под общ. ред. Л. Л. Рыбцовой. – М. : Юрайт, 2022. – 92 с. // Образовательная платформа «Юрайт» : сайт. – URL: <https://urait.ru/viewer/sovremennye-obrazovatelnye-tehnologii-493618#page/1> (дата обращения: 27.05.2025).

3. Куклина, Е. Н. Организация самостоятельной работы студента / Е. Н. Куклина, М. А. Мазниченко, И. А. Мушкина. – 2-е изд., испр. и доп. – М. : Юрайт, 2023. – 235 с. // Образовательная платформа «Юрайт» : сайт. – URL: <https://urait.ru/viewer/organizaciya-samostoyatelnoy-raboty-studenta-513809#page/1> (дата обращения: 27.05.2025).

ПРИЛОЖЕНИЕ А**Пример оформления титульного листа отчета**

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра автоматизации обработки информации (АОИ)

НАЗВАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Отчет по лабораторной работе по дисциплине

«Интерфейсы программных систем»

Вариант ____

Студент гр. ____

(И. О. Фамилия)

« ____ » _____ 20__ г.

Руководитель

ассистент каф. АОИ,

_____ Р. С. Кульшин

« ____ » _____ 20__ г.

Томск 20__