

МИНПРОСВЕЩЕНИЯ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
"Тульский государственный педагогический университет им. Л.Н. Толстого"
(ФГБОУ ВО "ТГПУ им. Л.Н. Толстого")

**МОДУЛЬ "РАЗРАБОТКА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ"**
Основы проектирования веб-интерфейсов

рабочая программа дисциплины (модуля)

Закреплена за кафедрой	институт передовых информационных технологий
ОПОП	Направление 02.03.02 Фундаментальная информатика и информационные технологии направленность (профиль) Инженерия программного обеспечения
Квалификация	Бакалавр
Год начала подготовки	2023
Форма обучения	очная
Общая трудоемкость	3 з.е.

Виды контроля по семестрам:
зачет 4

Семестр(Курс.Номер семестра на курсе)	4(2.2)		Итого	
	УП	РПД	УП	РПД
Лекции	18	18	18	18
Лабораторные	34	34	34	34
Итого ауд.	52	52	52	52
КСР	2	2	2	2
Контактная работа	54	54	54	54
Сам. работа	54	54	54	54
Часы на контроль	0	0	0	0
Практическая подготовка	0	0	0	0
Семинары	0	0	0	0
Консультации	0	0	0	0
Итого трудоемкость в часах	108	108	108	108

Программу составил(и):

доцент, Клепиков Алексей Константинович

Рабочая программа дисциплины

Основы проектирования веб-интерфейсов

разработана в соответствии с ФГОС:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии (приказ Минобрнауки России от 23.08.2017 г. № 808)

составлена на основании учебного плана:

Направление 02.03.02 Фундаментальная информатика и информационные технологии
направленность (профиль) Инженерия программного обеспечения
утвержденного Учёным советом вуза от 27.10.2022 протокол № 13.

РПД утверждена Учёным советом университета
протокол от 27.10.2022 г. № 13

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Достижение планируемых результатов обучения, соотнесенных с общими целями и задачами ОПОП.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Цикл (раздел) ООП:	Б1.В.01
2.1	Требования к предварительной подготовке обучающегося:
1.	Алгоритмы и структуры данных
2.	Веб-программирование
3.	Практикум по программированию
4.	Архитектура вычислительных систем
5.	Введение в специальность
6.	Вычислительные сети
7.	Программирование
8.	Введение в программирование
9.	Учебная практика по получению первичных навыков профессиональной деятельности
2.2	Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
1.	Компьютерное моделирование
2.	Методы оптимизации
3.	Разработка приложений для мобильных платформ
4.	Экономические информационные системы
5.	Информационная безопасность
6.	Применение нейронных сетей
7.	Разработка экономических информационных систем
8.	Тестирование программного обеспечения
9.	Экономико-правовые основы рынка программного обеспечения
10.	Аналитика больших данных
11.	Исследование операций
12.	Машинное обучение
13.	Проектирование и разработка пользовательских интерфейсов
14.	Производственная эксплуатационная практика

3. СООТНЕСЕНИЕ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ) С ИНДИКАТОРАМИ ДОСТИЖЕНИЯ КОМПЕТЕНЦИЙ**3.1 Компетенции обучающегося и индикаторы их достижения:**

ОПК-3: Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям

ОПК-3.1	Знает методы теории алгоритмов, методы системного и прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей возможности реализации интерфейсов в рамках различных операционных систем и браузеров
ОПК-3.2	Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем оценивать эффективность и удобство спроектированного веб-интерфейса
ОПК-3.3	Имеет практический опыт применения разработки программного обеспечения практический опыт разработки пользовательских веб-интерфейсов
ОПК-5: Способен устанавливать и сопровождать программное обеспечение информационных систем и баз данных, в том числе отечественного происхождения, с учетом информационной безопасности	
ОПК-5.1	Знает методику установки и администрирования информационных систем и баз данных. Знаком с содержанием Единого реестра российских программ возможности реализации интерфейсов в рамках различных операционных систем и браузеров.
ОПК-5.2	Умеет реализовывать техническое сопровождение информационных систем и баз данных оценивать эффективность и удобство спроектированного веб-интерфейса

ОПК-5.3	Имеет практические навыки установки и инсталляции программных комплексов, применения основ сетевых технологий
практический опыт разработки пользовательских веб-интерфейсов	
ПК-2: Способность к реализации всех этапов жизненного цикла программного обеспечения на основе современных парадигм, методологий, инструментальных и вычислительных средств	
ПК-2.1	Знает основы разработки и реализации процессов жизненного цикла программного обеспечения
ПК-2.2	Умеет разрабатывать и документировать программные средства на всех этапах их жизненного цикла
ПК-2.3	Имеет практический опыт разработки программных средств и документации
3.2 Результаты обучения по дисциплине:	
В результате освоения дисциплины обучающийся должен:	
	Знать:
3.1	возможности реализации интерфейсов в рамках различных операционных систем и браузеров.
	Уметь:
У.1	оценивать эффективность и удобство спроектированного веб-интерфейса
	Владеть:
В.1	практический опыт разработки пользовательских веб-интерфейсов

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература	Содержание
	Понятие интерфейса информационной системы				
1.1	Интерфейсы информационных систем. Классификация интерфейсов. Понятие пользовательского интерфейса. Проблемы интеграции компонентов ИС. /Лек/	4	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Интерфейс командной строки (Command Line Interface или CLI)</p> <p>Среди областей применения интерфейса командной строки можно выделить DOS-компьютеры. Взаимодействие происходит с помощью ввода команд. Компьютер обрабатывает эти команды и выводит на экран очередную строку. Данный тип UI давно устарел. Большинство CLI заменены графическими интерфейсами.</p> <p>Текстовый интерфейс пользователя (Text User Interface или TUI)</p> <p>Этот тип интерфейса пользователя предназначен для работы с символами. Исполнение происходит в режиме аппаратного текста, однако часто используется и дисплей. В данном случае на каждый источник у программиста имеется 256 символов. Навигация производится клавиатурой, а не мышью. В качестве примера можно привести Norton Commander или Turbo Pascal. Этот интерфейс также используется в загрузчиках ОС и BIOS-программах. Данный тип интерфейса также используется для установки операционных систем.</p>

1.2	История развития интерфейсов информационных систем /Лаб/	4	10	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Графический пользовательский интерфейс (GUI — Graphical User Interface) это средства позволяющие пользователям взаимодействовать с аппаратными составляющими компьютера достаточно комфортным и удобным для себя образом.</p> <p>В течении многих лет для большого количества операционных систем, таких как OS/2, Macintosh, Windows, AmigaOS, Linux, Symbian OS, и т. п., было создано еще большее количество графических интерфейсов.</p> <p>Давайте попробуем взглянуть на эволюцию дизайна интерфейсов этих систем, начиная с 80-х годов.</p> <p>Должен заметить, этот топик демонстрирует только значительные достижения и этапы в области графического дизайна (а не операционных систем в целом), да и не все системы существуют и по сей день.</p> <p>Первый GUI был разработан в Xerox Palo Alto Research Center (PARC) в далеких 70-х годах. Эта разработка дала старт новой эре инноваций в компьютерной графике.</p> <p>Первый персональный компьютер который использовал новый графический интерфейс был Xerox Alto, созданный в 1973 году. Он не был коммерческим продуктом и предназначался в основном для научных исследований в университетах.</p>
-----	--	---	----	----------------------------	---

1.3	Эволюция элементов пользовательского интерфейса /Ср/	4	8	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>При разработке первого графического пользовательского интерфейса за основу были взяты элементы реального мира: мусорная корзина, папка, изображение дискеты в качестве кнопки сохранения. Сегодня многие иконки считаются устаревшими, но все равно используются.</p> <p>Даже при использовании современных изображений и иконок дизайнеры стараются хотя бы минимально отразить их предназначение. Это позволяет облегчить интуитивное взаимодействие с интерфейсом. Цель GUI заключается в том, что люди могли легко определить предназначение каждой кнопки. Благодаря этому нам не приходится запоминать все команды, как это было в случае с командной строкой.</p> <p>Инструкции и правила</p> <p>При разработке GUI применяются определенные своды правил, которые помогают сделать программы удобнее в использовании. В качестве примера можно привести 8 золотых правил от Бена Шнайдермана. Ниже приведем несколько сносков из этих правил:</p> <p>Согласованность: взаимодействие должно происходить всегда похожим образом. То есть, следует избегать использования панелей управления с опциями типа “скопировать выделенную область”, “удалить выделенную область”, “добавить выделенную область”.</p> <p>Данный пример показывает отсутствие согласованности в GUI, чего следует избегать;</p> <p>Информативная обратная связь: все действия, производимые пользователем, должны быть подкреплены обратной связью. Например, если двойной клик открывает программу, то человеку приходится подождать пару секунд, прежде чем он сможет пользоваться этой программой. Чтобы пользователь знал, что его действия принесли результат, нужно проинформировать его об этом. Это можно реализовать сменой курсора. Один из старейших и привычных примеров – это курсор с песочными часами в Windows;</p> <p>Не перегружайте память пользователей: пользователи не в силах запомнить все и сразу. В длинных сегментах взаимодействия, где пользователь вынужден переходить по нескольким окнам, информация всегда должна отображаться в одной и той же области. Менее востребованная информация, которая отображалась в самом начале, должна быть скрыта.</p>
-----	--	---	---	----------------------------	--

1.4	Способы взаимодействия пользователя с интерфейсом. Компоненты интерфейса информационной системы. /Лек/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Тактильные интерфейсы пользователя (TUI или tangible user interface)</p> <p>В них взаимодействие происходит за счет применения мячей или других физических объектов. Сегодня данный тип интерфейсов редко используется в повседневной жизни. Если рабочий компьютер постоянно стоит на одном столе, применение тактильных интерфейсов приобретает новый смысл, однако чаще всего они просто неприменимы в повседневной жизни. Музеи и выставки – отличный пример сферы применения TUI.</p> <p>Физическое взаимодействие запоминается лучше любого другого. Кроме этого тактильные интерфейсы дают простор реализации объектов: форма, фактура, цвет. От песочницы с деревянными кубиками до увеличительного стекла для изображений – возможно практически все.</p> <p>Натуральный пользовательский интерфейс (NUI или natural user interface)</p> <p>Натуральный пользовательский интерфейс призван предоставить пользователю естественный и интуитивный опыт взаимодействия с устройством или программным обеспечением. В то же время, сам интерфейс будет видимым, например, на сенсорном экране. При помощи NUI команды пользователя вносятся с помощью жестов и прикосновений.</p> <p>Данный тип интерфейса пользователя также можно комбинировать с VUI. Благодаря прямому отклику устройства взаимодействие происходит естественней, нежели при вводе мышью или клавиатуры. Кроме сенсорных устройств NUI также можно использовать в игровых приставках.</p> <p>К примеру, Nintendo Wii позволяет воспроизводить действия на экране за счет перемещения контроллера рукой. Среди других примеров – дополнение Kinect к Xbox, которое позволяет управлять игровым персонажем на экране движениями собственного тела. Что делает взаимодействие более натуральным.</p>
-----	---	---	---	----------------------------	---

1.5	Стандарты в области разработки интерфейсов информационных систем /Ср/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Перцептивный пользовательский интерфейс (PUI или perceptual user interface)</p> <p>Перцептивный пользовательский интерфейс – интерфейс, управление которым происходит за счет восприятия человека. На сегодняшний день он до сих пор находится на стадии разработки. PUI, в теории, должен совмещать в себе возможности GUI и VUI, а также уметь распознавать жесты для взаимодействия с компьютером. Интеграция визуального и слухового восприятия жестов и звуков должно позволить PUI предоставить пользователям максимальный уровень восприятия и естественности.</p> <p>Интерфейс мозг-компьютер (BCI и brain-computer interface)</p> <p>Этот интерфейс пользователя в качестве источника команды использует человеческий мозг. На сегодняшний день эта технология достигла высокого уровня развития. Для измерения мозговых волн используются электроды, после чего полученная информация расшифровывается различными алгоритмами. Это и позволяет управлять роботизированными конечностями. Такой тип взаимодействия – большое преимущество для людей с ограниченными физическими возможностями.</p>
-----	---	---	---	----------------------------	--

1.6	Терминологический аппарат дисциплины /Ср/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Ценность оптимизации под поисковые системы</p> <p>В разработке графического интерфейса пользователя GUI и сайта есть как схожие моменты, так и различия. Например, посетитель пользуется навигацией по сайту. Он выбирает определенный путь сквозь структуру страниц. В графическом интерфейсе разработчик может контролировать, какие пункты будут доступны пользователю в тот или иной момент. Если функция недоступна, разработчик может скрыть эту опцию.</p> <p>В случае с сайтом у пользователя всегда есть возможность вернуться назад на страницу. Следовательно, навигацию также необходимо учитывать при создании сайта. Иерархия страниц должна быть максимально прозрачной и продуманной. Если ваш сайт состоит из нескольких уровней, то логично использовать навигацию типа “хлебные крошки”.</p> <p>Люди используют программы уже достаточно долгое время. Следовательно, мы уже привыкли к большинству стандартных элементов любого графического интерфейса. Сайты же появились относительно недавно.</p> <p>Веб-дизайнеры должны стараться продумать опыт взаимодействия с пользователем на максимальном уровне, и руководствоваться при этом проверенными практиками. Например, меню навигации лучше всего располагать в левом верхнем углу. Как вебмастер вы должны убедиться, что все элементы легко доступны любому посетителю. Это сделает ваш сайт удобным для использования.</p>
1.7	Компетенции и задачи специалистов в области проектирования интерфейса /Ср/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Диаграмма пользовательских маршрутов Сначала подумайте, как пользователи будут взаимодействовать с вашим приложением.</p> <p>Как пользователи могут попасть в нужный им раздел? Можно ли на диаграмме потоков визуализировать все возможные маршруты пользователей по приложению?</p> <p>Важно максимально всё упростить, уменьшив беспорядок и удалив ненужные шаги. Это действительно поможет на начальном этапе наполнить ваш UI-дизайн.</p> <p>Для удобства можно использовать различные готовые инструментари, например, шаблоны блок-схем Miro, Milanote и Whimsical. Можно легко формировать маршруты с помощью трёх стандартных фигур:</p> <ul style="list-style-type: none"> прямоугольник — для представления изображений на экране; ромб — для представления доступных пользователю решений; стрелка — для отображения связей между экранами и решениями.

	Проектирование интерфейсов информационных систем на основе анализа пользовательского поведения				
2.1	Основные принципы разработки интерфейсов информационных систем. Понятие UI/UX-дизайна /Лек/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Изучение дизайнерских шаблонов и стилей Планирование облегчает реализацию, поэтому всегда делите задачи на более мелкие части.</p> <p>Рутинная работа дизайнера и их повседневные инструменты могут привести к бездумному копированию шаблонов, без творческого подхода и индивидуальности.</p> <p>Дорабатывайте и совершенствуйте их, чтобы сделать ваше приложение как можно лучше.</p> <p>Отличным источником информации будет UI Patterns. Это прекрасный ресурс, чтобы ознакомиться с наиболее часто используемыми шаблонами проектирования и способами их применения.</p>
2.2	Инструменты и методы проектирования интерфейса. Этапы проектирования. Целеполагание в дизайне интерфейса. /Лаб/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Создание каркасов На этом этапе уже можно переходить к творчеству. Каркасы являются важной частью в проектировании, поэтому их никогда не следует пропускать.</p> <p>Хорошо, если вы знакомы с инструментом для создания прототипов. Он поможет создать лучший дизайн для приложения. Но если для вас это в новинку, вы, скорее всего, захотите вернуться к классическому варианту: ручка и бумага. Освоения новых инструментов только замедлит вас, поэтому не бойтесь использовать простые вещи вроде ручки и бумаги.</p> <p>Обратите внимание на SneakPeekIt. В нём есть большое количество шаблонов, которые можно распечатать и использовать в своих эскизах. Это совершенно бесплатный инструмент для вдохновения.</p>
2.3	Возможности проектирования интерфейсов средствами онлайн-приложений. Работа с технической документацией. /Ср/	4	8	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Создание макетов приложения Как только иерархия установлена и структура каждого экрана определена, вы можете перейти к программному обеспечению, например, Figma или Sketch, для разработки своего продукта. Figma — облачный инструмент для UI-дизайна. Идеально подходит для реализации каркасов на этом этапе процесса.</p>

2.4	Виды документации в проектировании интерфейса /Лек/	4	2	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p style="text-align: center;">Создание мокапа</p> <p>Этот этап позволяет быстро понять видение клиента и внести уйму изменений до начала разработки интерфейса пользователя. Мы намечаем расположение кнопок, форм и других нужных элементов, а уже потом подбираем цветовую палитру, шрифты, изображения, преобразуя всё это в удобный и красивый макет. То есть начинаем с варфрейма (план расположения элементов на странице), а заканчиваем созданием из этого плана красивой картинки. В случае разработки приложений под Android и iOS труд дизайнера облегчается гайдлайнами — правилами оформления и расположения элементов интерфейса, регламентом UX/UI, который был создан непосредственно экспертами по дизайну из Google и Apple.</p>
	Проектирование пользовательского опыта и прототип интерфейсного решения				
3.1	Понятие пользовательского опыта и информационная система. Адаптивный и отзывчивый дизайн. /Лек/	4	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Для того, чтобы понять, что такое UX (User Experience, пользовательский опыт), обратимся к словам Дона Нормана, соучредителя и консультанта Nielsen Norman Group и автора нескольких книг о пользовательском опыте. Он рассказывает о том, как он довольно давно — когда компьютеры только начали появляться в продаже — работал в Apple, и упоминает, что они в тот момент думали о пользовательском опыте: они говорили о том, что опыт — это когда человек впервые узнает о компьютере, видит его в магазине, покупает его, вздыхает, потому что не может уместить эту большую коробку в машине, привозит домой, открывает и смотрит на него испуганными глазами, не понимая, что с ним делать дальше. Все перечисленные моменты тогда считались пользовательским опытом, потому что это все — точки соприкосновения с продуктом, и каждый из этих моментов влияет на пользовательский опыт. Любой опыт взаимодействия пользователя с продуктом является пользовательским опытом.</p>

3.2	Адаптивный и отзывчивый дизайн. Разработка анимированного прототипа интерфейса. /Лаб/	4	6	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Чего же ждут работодатели от продактов по части UX? Из поиска вакансий можно сделать вывод, что чаще всего встречаются следующие требования: понимание и навыки в UX; опыт в UX, работа с продуктами и мобильными приложениями, критичными к UX; разработка UX-прототипов. Если переводить эти требования на язык, которым можно общаться с продактами, то получается следующая картина:</p> <p>Если продакт попадает в малую команду, где он — «человек-оркестр» и должен делать чуть ли не все подряд, то от него ожидается, что он, во-первых, сможет сделать прототип интерфейса, а во-вторых, имеет какой-то опыт касательно разных интерфейсов и способен предлагать разумные решения.</p> <p>Если команда большая и в ней присутствуют отдельные роли UX-дизайнера, UX-исследователя и так далее, то от продакта ожидается, что он: не будет впадать во вкусовщину и мешать дизайнерам делать какие-то предложения по изменению продукта; сможет организовать UX-исследование и сделать из него какие-то выводы; будет преследовать цели бизнеса. На последнем пункте стоит в особенности заострить внимание — удобство пользователя, конечно, тоже является очень важным, но у бизнеса есть какие-то цели, которые не всегда совпадают с интересами пользователей.</p>
3.3	Применение метафор и аффорданса в проектировании интерфейса. Инструменты прототипирования интерфейса. Создание прототипа интерфейса. /Ср/	4	14	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Самой важной из этих ценностей является не интерфейс или что-то еще, а безопасность. Как правило, при запуске нового продукта первые разговоры идут именно о безопасности. Например, сейчас планируется введение электронных трудовых книжек, и, согласно опросам, наибольшее число потенциальных пользователей беспокоит именно вопрос безопасности — они боятся, что данные могут быть утеряны. Поэтому без решения вопроса безопасности продукта остальные его составляющие будут попросту бессмысленны. Касательно безопасности также появляется наибольшее количество слухов при появлении нового продукта на рынке, примерами могут служить статьи о вреде микроволновок, в свое время имевшие огромную популярность, или вездесущие слухи о вреде 5G, распространяемые прямо сейчас. Только когда вопрос с безопасностью оказывается закрыт, пользователи начинают интересоваться другими аспектами продукта.</p>
	Визуальные компоненты дизайна интерфейса				

4.1	Особенности морфологии, композиции, эргономики интерфейса информационной системы. Виды визуальных компонентов интерфейса информационной системы. Понятие визуальной иерархии. /Лек/	4	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Следующим по важности аспектом является стабильность. Если мы говорим о цифровых продуктах, то они должны стабильно работать, чтобы от них можно было получить приятный пользовательский опыт. На эту тему есть исследования — например, если посмотреть по периодам, когда Facebook работает нестабильно или к нему вовсе пропадает доступ, то можно заметить, что в это же время аудитория Telegram растет: от сломанного Facebook людям не нужен никакой пользовательский опыт и они начинают пользоваться другим продуктом для получения опыта общения.</p> <p>Следующий аспект — скорость взаимодействия с продуктом, скорость его работы. Это тоже довольно важный момент, каким бы красивым и удобным продукт ни был, вы не станете с ним работать, если он сильно тормозит.</p> <p>Далее — «паттерность». Под «паттерностью» понимается то, насколько продукт интуитивно понятен пользователю, насколько он соответствует общепринятым интерфейсным и поведенческим паттернам (отсюда и название). Возможно, вы решили проявить креативность и сделали что-то новое в своем продукте. А пользователь, в свою очередь, начинает испытывать трудности по поводу того, как с этим продуктом взаимодействовать, потому что он не похож ни на какой из продуктов со схожим функционалом из тех, которыми он пользовался ранее.</p>
4.2	Особенности морфологии, композиции, эргономики интерфейса информационной системы. Виды визуальных компонентов интерфейса информационной системы. Понятие визуальной иерархии. Современные тенденции в развитии дизайна интерфейса. Гайдлайны. Понятие Material Design. Анализ визуальной иерархии и компонентов интерфейса информационной системы. Юзабилити-аудит сайта. /Ср/	4	18	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	<p>Следующий аспект — сервисность. Сервисность — это все, что находится вокруг продукта: доставка, техподдержка и так далее, то есть все то, что продуктом не является, но при этом влияет на пользовательский опыт, который человек получит от взаимодействия с продуктом. Например, если вам очень понравился интернет-магазин, продающий, допустим, шины, и прямо там же вы можете записаться на шиномонтаж, но при этом когда вы приезжаете на шиномонтаж, вам там хамят, а шины ставят откровенно халтурно — какой у вас будет пользовательский опыт по итогу? Наверняка вы вряд ли обратитесь туда еще раз. И так же со многими другими продуктами.</p> <p>Следующие вещи, которые уже не так сильно влияют на пользовательский опыт, это «модность» (насколько продукт отвечает современным тенденциям по части интерфейса, поведения и так далее) и «восхитительность» (какие-то фишки внутри продукта, способные вызвать у пользователя восхищение).</p>

4.3	Разработка дизайна интерфейса для регистрации пользователя в информационной системе (sign up). Разработка дизайна интерфейса приложения на примере калькулятора. /Лаб/	4	10	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	Эта иерархия является подспорьем для приоритизации задач: если задачи, касающиеся более важных аспектов продукта, решены плохо или не решены вообще, то нет смысла заморачиваться над менее важными аспектами: нет смысла добавлять фичи, призванные вызывать у пользователя восхищение, если продукт банально тормозит. Если вы, допустим, тратите 80% усилий на «модность» и «восхитительность», то результат не будет хоть сколько-нибудь удовлетворительным. Если же вы достигли хорошей реализации, то уже тогда можно приступить к наименее важным аспектам. Наибольшая часть усилий должна быть приложена к «фундаменту» продукта.
4.4	Разработка визуальных компонентов дизайна пользовательского интерфейса /Лаб/	4	6	Л1.1 Л1.2 Л1.3Л2.1 Л2.2	Для того, чтобы полноценно понять, что такое пользовательский интерфейс, нужно представить следующую ситуацию. Допустим, есть пользователь и у него есть цель, которой он хочет достичь с помощью нашего продукта. Пользовательский интерфейс — это одновременно и дорога от пользователя до цели, и препятствие между пользователем и целью. Из этого утверждения можно сделать вывод, что идеальный пользовательский интерфейс — это интерфейс, которого нет, но при этом все его функции выполняются. Например, у вас есть потребность похудеть. Вы скачиваете приложение, которое помогает вам с самоконтролем. Вы худеете месяц или два, приложение привело вас к цели и вы похудели, при этом интерфейс был максимально удобным для вас. И вдруг на рынке появляется другой продукт. Ваша цель не изменилась — похудеть, но при этом вам предлагают, условно говоря, установить в голову чип, который будет точно так же помогать вам с самоконтролем, но при этом вы вообще не будете об этом думать и добьетесь нужного результата за более короткий промежуток времени. Если вопрос с безопасностью этого чипа решен, то вы, скорее всего, выберете второй вариант, потому что это более короткий путь к вашей цели. Несмотря на все преимущества приложения, которым вы пользовались до этого, оно является более длинной дорогой между пользователем и целью. Чем короче получится эта дорога — тем лучше для пользователя.
4.5	/КСР/	4	2		КСР

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Типовые задания для проведения текущего контроля

1 Расшифруйте аббревиатуру DOM

- Document Object Model
- Digital Optical Modulation
- Domestic Object Mode

2 Что такое hoisting в JavaScript?

- Это механизм в JavaScript, в котором переменные и объявления функций передвигаются вверх своей области видимости перед тем, как код будет выполнен
- Это объект, который хранит все переменные и функции
- Это услуга по предоставлению ресурсов для размещения информации на сервере, постоянно находящемся в сети

3 Какие ключевые слова используются в JavaScript для того, чтобы объявить переменную?

- var, let, const
- byte, short, int, long, float
- int, short, var

4 Расшифруйте аббревиатуру API

- Analog Programm Interface
- Application Programming Interface
- Academy Provide Infinite

5 Какое API предоставляет клиенту функциональность для обмена данными между клиентом и сервером? Данный - API предоставляет простой способ получения данных по ссылке без перезагрузки страницы.

- PerformancePaintTiming
- NavigatorStorage
- XMLHttpRequest

6 Этот метод разбивает строку на подстроки, используя заданный разделитель, и возвращает их в виде массива.

- filter()
- split()
- parseInt()

7 Метод, который возвращает строку, где все буквенные символы преобразованы в верхний регистр с учетом текущего языкового стандарта среды размещения.

- toLocaleUpperCase()
- toLowerCase()
- toString()

8 Метод, который округляет аргумент до ближайшего большего целого.

- Math.asin()
- Math.ceil()
- Math.max()

9 Как в JavaScript создать массив?

- var array = new Array() или var array = []
- var array = new Array{ } или var new array = []
- int new Array() или var new Array()

10 Как в JavaScript можно вызвать функцию?

- function myFun()
- myFun()
- myFun

11 Что такое замыкание в JavaScript?

- Это способность функции - вызывать другую функцию после завершения своей работы
- Это способность функции - запоминать все переменные
- Это способность функции - запоминать область видимости, в которой эта функция была объявлена

12 Расшифруйте аббревиатуру DOM

- Document Object Model
- Digital Optical Modulation
- Domestic Object Mode

13 Что такое hoisting в JavaScript?

- Это механизм в JavaScript, в котором переменные и объявления функций передвигаются вверх своей области видимости перед тем, как код будет выполнен
- Это объект, который хранит все переменные и функции
- Это услуга по предоставлению ресурсов для размещения информации на сервере, постоянно находящемся в сети

14 Какие ключевые слова используются в JavaScript для того, чтобы объявить переменную?

- var, let, const
- byte, short, int, long, float
- int, short, var

15 Расшифруйте аббревиатуру API

- Analog Programm Interface
- Application Programming Interface
- Academy Provide Infinite

16 Чем отличается const от let?

- const - не является частью JavaScript
- Переменные, объявленные через const, находятся в глобальной видимости
- Объявление const задаёт константу, то есть значение, которое нельзя менять

17 Что такое рекурсия в JavaScript?

- Рекурсия — это метод программирования, позволяющий функции вызывать саму себя
- Рекурсия — это метод программирования, позволяющий функции быть универсальной
- Рекурсия не является частью JavaScript

18 Расшифруйте аббревиатуру ES

- EnterSafe
- ECMAScript
- EffectService

5.2. Типовые задания для проведения промежуточной аттестации

Вопросы к зачету

1. Понятие пользовательского интерфейса.
2. Понятие UX и UI.
3. Понятие модели поведения пользователя.
4. Отличия между принципами построения интерфейса.
5. Критерии оценок пользовательского интерфейса.
6. Документирование интерфейса.
7. Элементы интерфейса системы.
8. Принцип построения прототипов сайта.
9. Figma как инструмент для проектирования интерфейса.
9. Sketch как инструмент для проектирования интерфейса.
10. Тестирование прототипов.
11. Сценарии поведения пользователей.
12. Модели поведения пользователей в зависимости от используемого устройства.
13. XML и XSLT.
14. Построение логики взаимодействия интерфейса приложения и серверной части.
15. Понятие и принципы использования API.
16. Управление содержимым интерфейса с помощью языка javascript

5.3. Перечень видов оценочных средств

Зачет
Тест
Индивидуальные проекты
Лабораторные работы

5.4. Процедура применения оценочных материалов

Промежуточная аттестация может проводиться с применением электронного обучения и (или) дистанционных образовательных технологий в соответствии с "Порядком проведения промежуточной аттестации с применением электронного обучения и /или дистанционных образовательных технологий".

Для получения положительной итоговой оценки на зачете необходимо получить не менее 50% по каждой составляющей и выполнить все практические задания.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год (кол-во экземпляров для печатных изданий)	Ссылка на электронное издание
Л1.1	Павловская Е. Э.	Графический дизайн. Современные концепции: Учебное пособие	, 2017	http://www.biblio-online.ru/book/5CF926E6-F85E-4BC7-8AA5-1F51608D8883
Л1.2	Терещенко П. В., Астапчук В. А.	Интерфейсы информационных систем: учебное пособие	, 2012	http://biblioclub.ru/index.php?page=book&id=228775

	Авторы, составители	Заглавие	Издательство, год (кол-во экземпляров для печатных изданий)	Ссылка на электронное издание
Л1.3	Астапчук В. А., Терещенко П. В.	Корпоративные информационные системы: требования при проектировании: Учебное пособие	, 2019	https://www.biblio-online.ru/book/korporativnye-informacionnye-sistemy-trebovaniya-pri-proektirovanii-444114

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год (кол-во экземпляров для печатных изданий)	Ссылка на электронное издание
Л2.1	Брокшмидт К.	Пользовательский интерфейс приложений для Windows 8, созданных с использованием HTML, CSS и JavaScript: учебный курс	, 2016	http://biblioclub.ru/index.php?page=book&id=429247
Л2.2		Графический интерфейс, пакеты приложений	, 2004 (11 шт.)	

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Онлайн-редактор Figma
Э2	Интуит. Национальный открытый университет
Э3	UX Design for Mobile Developers // Udacity

6.3. Информационные технологии

6.3.1 Перечень лицензионного и свободно распространяемого программного обеспечения

1.	Операционная система Microsoft Windows XP Professional Russian. Лицензия № 16698685 от 08.08.2003 г.
2.	Операционная система Microsoft Windows Professional 7 Russian. Лицензия №48497058 от 13.05.2011 г., договор № Пр/16/6 от 05 апреля 2016 г.
3.	Операционная система Microsoft Windows 10 Professional Russian. Контракт № ПР/ФЕН/15/18 от 23.10.2015 г., договор № Пр/16/6 от 05 апреля 2016 г.
4.	Программное обеспечение Microsoft Office Enterprise 2007 Russian. Лицензия №46138962 от 16.11.2009
5.	Программное обеспечение Microsoft Office 2013 Professional. Контракт № 405535 от 2 ноября 2015 года, контракт № ПР/ФЕН/15/18 от 23.10.2015 г.
6.	Комплексная система антивирусной защиты Kaspersky Endpoint Security для бизнеса – стандартный Russian Edition. 500-999 Node 2 year Educational Renewal License. Лицензия № 13С8-190514-084943-783-1256 от 15.05.2019
7.	Браузеры Google Chrome, Mozilla, Opera. Свободно распространяемое ПО
8.	Текстовый редактор NotePad++. Свободно распространяемое ПО
9.	Среда программирования и набор инструментов для программирования. MinGW 0.6.3 Свободно распространяемое ПО

6.3.2 Перечень информационных справочных систем и профессиональных баз данных

1.	Официальный интернет-портал базы данных правовой информации (http://pravo.gov.ru)
2.	Портал Федеральных государственных образовательных стандартов высшего образования (http://fgosvo.ru)

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Ауд.	Назначение	Оборудование и технические средства обучения	Вид
2-15	Компьютерный класс	компьютеры, рулонный экран, стол преподавателя, столы компьютерные, переносной проектор	Лаб
2-16	Компьютерный класс	интерактивная доска, компьютеры, маркерная доска, принтер, сканер, стол преподавателя, столы учебные	Лаб
4-318	Компьютерный класс	компьютеры, маркерная доска, серверная стойка лаборатории МТС, стол преподавателя, столы компьютерные, столы учебный большой	Лаб
4-305	Компьютерный класс	аудиоколонки для проектора и интерактивной доски, аудиоколонки учебные, интерактивная доска, компьютеры, кондиционер, маркерная доска, проектор, столы компьютерные, столы учебные	Лек
4-305	Компьютерный класс	аудиоколонки для проектора и интерактивной доски, аудиоколонки учебные, интерактивная доска, компьютеры, кондиционер, маркерная доска, проектор, столы компьютерные, столы учебные	Лаб

Ауд.	Назначение	Оборудование и технические средства обучения	Вид
4-306	Компьютерный класс	аудиоколонки для проектора и интерактивной доски, интерактивная доска, компьютеры, кондиционер, маркерная доска, проектор, столы компьютерные, столы учебные	Экзамен
4-306	Компьютерный класс	аудиоколонки для проектора и интерактивной доски, интерактивная доска, компьютеры, кондиционер, маркерная доска, проектор, столы компьютерные, столы учебные	Лек
4-307	Компьютерный класс	аудиоколонки, компьютеры, кондиционер, маркерная доска, столы компьютерные, столы учебные, телевизор	КСР
4-307	Компьютерный класс	аудиоколонки, компьютеры, кондиционер, маркерная доска, столы компьютерные, столы учебные, телевизор	
4-318	Компьютерный класс	компьютеры, маркерная доска, серверная стойка лаборатории МТС, стол преподавателя, столы компьютерные, столы учебный большой	КР
4-303	Помещение для самостоятельной работы	аудиоколонки, кондиционер, маркерная доска, столы компьютерные, столы учебные, компьютерная техника с возможностью подключения сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета	Ср

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

При изучении дисциплины целесообразно выполнять следующие рекомендации:

- а) изучение должно быть систематическим, в течение всего семестра; для формирования устойчивых навыков и знаний эффективнее следовать логике построения курса и осваивать учебный материал небольшими порциями;
- б) после изучения темы на практическом занятии следует выполнить самостоятельную работу по данной теме, если она предполагается, а также изучить дополнительные ресурсы, рекомендованные преподавателем, а также найденные самостоятельно в сети Интернет, в фондах научной библиотеки университета.

Дополнительные учебные результаты могут быть достигнуты в форме научно-исследовательской работы студентов по проблематике данной учебной дисциплины: написание научных статей, выступление на конференциях, круглых столах, участие в научно-исследовательских проектах. Выполнение практических заданий предусмотрено как на практических занятиях, так и в форме самостоятельной работы. При выполнении самостоятельной работы учащийся может задавать уточняющие вопросы преподавателю.