|  |  |
| --- | --- |
|  |  |

**КОНКУРСНОЕ ЗАДАНИЕ КОМПЕТЕНЦИИ**

**«Разработка мобильных приложений»**

**Финала Чемпионата по профессиональному мастерству «Профессионалы» в 2025 г.**

**г. Нижний Новгород**

2025 г.

Конкурсное задание разработано экспертным сообществом и утверждено Менеджером компетенции, в котором установлены нижеследующие правила и необходимые требования владения профессиональными навыками для участия в соревнованиях по профессиональному мастерству.

**Конкурсное задание включает в себя следующие разделы:**

[1. ОСНОВНЫЕ ТРЕБОВАНИЯ КОМПЕТЕНЦИИ…………………………….4](#_Toc142037183)

[1.1. Общие сведения о требованиях компетенции……………………………...4](#_Toc142037184)

[1.2. Перечень профессиональных задач специалиста по компетенции «Разработка мобильных приложений»…………………………………………..4](#_Toc142037185)

[1.3. Требования к схеме оценки………………………………………………...25](#_Toc142037186)

[1.4. Спецификация оценки компетенции………………………………………26](#_Toc142037187)

[1.5. Конкурсное задание………………………………………………………...26](#_Toc142037188)

[1.5.1. Разработка/выбор конкурсного задания…………………………………26](#_Toc142037189)

[1.5.2. Структура модулей конкурсного задания (инвариант/вариатив)……...27](#_Toc142037190)

[2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ…………………………...70](#_Toc142037191)

[2.1. Личный инструмент конкурсанта………………………………………….71](#_Toc142037192)

[2.2.Материалы, оборудование и инструменты, запрещенные на площадке…71](#_Toc142037193)

[3. ПРИЛОЖЕНИЯ……………………………………………………………….73](#_Toc142037194)

1.ОСНОВНЫЕ ТРЕБОВАНИЯКОМПЕТЕНЦИИ

1.1. Общие сведения о требованиях компетенции

Требования компетенции (ТК) «Разработка мобильных приложений» определяют знания, умения, навыки и трудовые функции, которые лежат в основе наиболее актуальных требований работодателей отрасли.

Целью соревнований по компетенции является демонстрация лучших практик и высокого уровня выполнения работы по соответствующей рабочей специальности или профессии.

Требования компетенции являются руководством для подготовки конкурентоспособных, высококвалифицированных специалистов / рабочих и  участия их в конкурсах профессионального мастерства.

В соревнованиях по компетенции проверка знаний, умений, навыков и  трудовых функций осуществляется посредством оценки выполнения практической работы.

Требования компетенции разделены на четкие разделы с номерами и  заголовками, каждому разделу назначен процент относительной важности, сумма которых составляет 100.

1.2. Перечень профессиональных задач специалиста по компетенции «Разработка мобильных приложений»

Перечень видов профессиональной деятельности, умений, знаний, профессиональных трудовых функций специалиста базируется на требованиях современного рынка труда к данному специалисту.

Таблица 1

**Перечень профессиональных задач специалиста**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Критерий / Модуль** | **Важность в %** |
| **1** | **Организация и управление работой** | **9,1** |
| Специалист должен знать и понимать:   * Алгоритмы решения типичных задач, области и способы их применения * Информационно-справочный и информационно-поисковый аппарат документа. * Информационно-справочный и поисковый аппарат документа. * Методика и стиль изложения документации пользователя (технических средств, программных средств). * Методы автоматической и автоматизированной проверки работоспособности компьютерного программного обеспечения. * Методы и приемы алгоритмизации поставленных задач * Методы и приемы формализации поставленных задач * Методы обработки первичной информации/источников первичной информации. * Методы оценки качества продукции в области информационных технологий. * Методы работы с базами данных. * Методы тестирования ИТ-продуктов. * Нормативно-правовая база применения стандартов. * Нотации и программное обеспечение для графического отображения алгоритмов * Область применения инструментальных средств для выполнения тестирования ПО. * Общие требования к структуре разделов технического документа. * Основная терминология по тестированию ПО * Основные актуальные средства генерации тестовых данных и области их применения. * Основные виды авторской разметки текста технической документации. * Основные виды диагностических данных проверки работоспособности компьютерного программного обеспечения и способы их представления. * Основные возможности современных текстовых процессоров. * Основные графические форматы и их особенности. * Основные методы измерения и оценки характеристик компьютерного программного обеспечения. * Основные способы работы с векторной и растровой графикой, способы включения рисунков в документ, правила оформления рисунков. * Основные стандарты оформления текстовых документов. * Основные стандарты оформления технической документации на компьютерное программное обеспечение. * Основные стандарты эксплуатационной документации, в том числе документации пользователя. * Основные термины и сокращения, используемые в технической документации и принятые в организации. * Основные типы эксплуатационных документов, адресованных пользователям, их особенности. * Основные форматы электронных документов и особенности их использования. * Основы верстки с использованием языков разметки. * Основы типографики. * Особенности основных операционных систем. * Перечень лидирующих инструментальных средств, их назначение, основные функциональные возможности, сильные и слабые стороны, способы применения: средства для набора текста (текстовый процессор, XML-редактор), средства подготовки снимков экрана, средства преобразования документов в выходные форматы, тестовый стенд. * Понятия «техническое средство», «программное средство», «комплекс», «система», содержание этих понятий, различия между ними. * Порядок проектирования, производства, поставки и внедрения, применения, эксплуатации, утилизации документируемой продукции. * Процедуры обеспечения безопасности при выполнении тестирования ПО. * Современное состояние индустрии информационных технологий, основные подходы и тенденции. * Стандарты документирования промышленной продукции, программных средств, систем (в том числе автоматизированных). * Терминология, применяемая для описания интерфейса пользователя компьютерных систем. * Типовые метрики компьютерного программного обеспечения. * Требования по обеспечению безопасности аппаратных и программных средств автоматизированных систем, используемых при выполнении тестовых процедур, включая вопросы антивирусной защиты. * Форматы представления тестовых данных. * Человеко-ориентированный подход к проектированию интерактивных систем. * Язык, на котором написана техническая документация тестируемого ПО, на уровне, достаточном для чтения технической документации * Языки формализации функциональных спецификаций * Языки, утилиты и среды программирования и средства пакетного выполнения процедур. |  |
| Специалист должен уметь:   * Анализировать исходную документацию в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Анализировать функциональные разрывы в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Выбирать необходимые генераторы тестовых данных. * Выполнять базовую настройку операционных систем. * Выявлять недостающую информацию для выполнения тестирования ПО в заданном объеме. * Использовать выбранную среду программирования для разработки процедур проверки работоспособности компьютерного программного обеспечения. * Использовать методы и приемы алгоритмизации поставленных задач. * Использовать методы и приемы формализации поставленных задач. * Использовать программное обеспечение для графического отображения алгоритмов. * Исследовать программные средства на тестовом стенде. * Исследовать продукт или технологию на тестовом стенде. * Исследовать техническую документацию, извлекать из нее сведения, необходимые для решения поставленной задачи. * Компоновать технический документ на основе заданных источников. * Опрашивать экспертов и анализировать полученные сведения. * Осуществлять коммуникации с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Оформлять рисунки, в том числе снимки экрана, оформлять подписи к ним в соответствии с используемым стандартом. * Оценивать количество рабочих часов, необходимых для выполнения полученного задания. * Оценивать количество рабочих часов, необходимых для выполнения полученного задания. * Оценивать соответствие ИТ-продуктов предъявляемым к ним требованиям. * Писать программный код процедур проверки работоспособности компьютерного программного обеспечения на выбранном языке программирования. * Подготавливать графические материалы в программах подготовки векторных изображений. * Подготавливать графические схемы. * Подготавливать необходимые средства и ресурсы для выполнения задания по тестированию ПО. * Подготавливать протоколы мероприятий в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Получать замечания у экспертов и вносить исправления в документ. * Получать и использовать информацию, необходимую для выполнения задания на тестирование ПО. * Преобразовывать технический документ в различные выходные форматы (PDF, HTML, формат электронной справки). * Применять алгоритмы решения типовых задач в соответствующих областях. * Применять генераторы тестовых данных. * Применять заданные стандарты и шаблоны для составления и оформления технической документации. * Применять требования используемых в проекте стандартов с учетом особенностей данного проекта. * Проводить интервью с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить консультации для разработчиков требований к ИТ-продуктам. * Проводить переговоры с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить презентации заинтересованным сторонам в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить ручное тестирование ИТ-продуктов. * Работать в современном текстовом процессоре. * Работать с инструментами подготовки тестовых данных. * Работать с макетами интерфейса пользователя программного средства на уровне текста, работать с ресурсными строками интерфейса пользователя программного средства. * Работать с типовой ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Разрабатывать документы в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Разрабатывать руководство по установке прикладного программного средства . * Разрабатывать руководство по эксплуатации бытового прибора. * Разрабатывать руководство пользователя прикладного программного средства. * Разрабатывать технологическую инструкцию для персонала автоматизированной системы. * Разрабатывать учебное пособие по прикладному программному средству. * Создавать в тексте якоря и гипертекстовые ссылки, оформлять подписи к гипертекстовым ссылкам. * Создавать графические схемы, получать снимки экрана, включать рисунки в технический документ и оформлять их * Создавать информационно-поисковый аппарат документа с помощью текстового процессора. * Создавать, настраивать, применять стили в документе с помощью текстового процессора. * Составлять календарный план выполнения полученного задания. * Составлять отчет о подготовке тестовых данных. * Составлять отчет о результатах подготовки к выполнению тестирования ПО. * Составлять текст, адаптированный для автоматизированного перевода. * Составлять требования к эксплуатационному документу. * Устанавливать корректную последовательность операций при выполнении тестирования ПО. * Устанавливать операционные системы. |
| **2** | **Проектирование** | **32,8** |
| Специалист должен знать и понимать:   * Алгоритмы решения типичных задач, области и способы их применения * Архитектура, устройство и функционирование вычислительных систем. * Виды технических носителей информации, правила их хранения и эксплуатации. * Возможности типовой ИС. * Действующие системы счислений, шифров и кодов. * Инструменты и методы выявления требований. * Инструменты и методы моделирования бизнес-процессов в ИС. * Инструменты и методы модульного тестирования. * Инструменты и методы прототипирования пользовательского интерфейса. * Инструменты и методы тестирования нефункциональных и функциональных характеристик ИС. * Источники информации, необходимой для профессиональной деятельности при выполнении работ по созданию (модификации) и сопровождению ИС. * Коммуникационное оборудование. * Культура речи. * Лучшие практики создания (модификации) и сопровождения ИС в экономике. * Математическая статистика. * Методы и приемы алгоритмизации поставленных задач * Методы и приемы формализации поставленных задач * Методы оценки качества продукции в области информационных технологий. * Методы представления статистической информации. * Методы проведения расчетов и вычислительных работ. * Методы проектирования механизированной и автоматизированной обработки информации. * Методы расчета выполненных работ. * Методы тестирования ИТ-продуктов. * Нотации и программное обеспечение для графического отображения алгоритмов * Общие принципы анимации. * Основные формализованные языки программирования. * Основы архитектуры мультиарендного (multitenancy) программного обеспечения. * Основы бухгалтерского учета и отчетности организаций. * Основы верстки с использованием языков описания стилей. * Основы верстки с использованием языков разметки. * Основы ИБ организации. * Основы маркетинга. * Основы международных стандартов финансовой отчетности. * Основы налогового законодательства Российской Федерации. * Основы организации производства. * Основы программирования с использованием сценарных языков. * Основы программирования. * Основы психологии. * Основы современных операционных систем. * Основы современных СУБД. * Основы управления организационными изменениями. * Основы управления персоналом, включая вопросы оплаты труда. * Основы управления торговлей, поставками и запасами. * Основы управленческого учета. * Основы экономики, организации труда и производства, правила и нормы охраны труда. * Основы эргономики в части создания систем индикации. * Отраслевая нормативно-техническая документация. * Правила деловой переписки. * Правила написания интерфейсных текстов. * Правила перспективы, колористики, композиции, светотени и изображения объема. * Правила типографского набора текста и верстки. * Правила типографского набора текста. * Предметная область автоматизации. * Программные средства и платформы инфраструктуры информационных технологий организаций. * Рабочие программы, инструкции, макеты и другие руководящие материалы, определяющие последовательность и технику выполнения расчетных операций. * Сетевые протоколы. * Системы классификации и кодирования информации, в том числе присвоения кодов документам и элементам справочников. * Системы хранения и анализа баз данных. * Современные методики тестирования разрабатываемых ИС. * Современные объектно-ориентированные языки программирования. * Современные подходы и стандарты автоматизации организации (CRM, MRP, ERP, …, ITIL, ITSM). * Современные стандарты информационного взаимодействия систем. * Современные структурные языки программирования. * Средства вычислительной техники, сбора, передачи и обработки информации и правила их эксплуатации. * Стандарты, регламентирующие требования к эргономике взаимодействия человек - система. * Тенденции в графическом дизайне. * Теория баз данных. * Теория цвета. * Техники и методики подготовки графических материалов. * Технические требования к интерфейсной графике. * Технологии алгоритмической визуализации данных. * Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии. * Технологии подготовки и проведения презентаций. * Технологию механизированной и автоматизированной обработки информации. * Требования и руководства по проектированию платформ и операционных систем. * Требования целевых операционных систем и платформ к пиктограммам и элементам управления. * Устройство и функционирование современных ИС. * Человеко-ориентированный подход к проектированию интерактивных систем. * Языки программирования и работы с базами данных. * Языки современных бизнес-приложений. * Языки формализации функциональных спецификаций |  |
| Специалист должен уметь:   * Анализировать исходную документацию в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Анализировать функциональные разрывы в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Верстать текст. * Выполнять верстку. * Использовать методы и приемы алгоритмизации поставленных задач. * Использовать методы и приемы формализации поставленных задач. * Использовать программное обеспечение для графического отображения алгоритмов. * Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Оптимизировать интерфейсную графику под различные разрешения экрана. * Осуществлять коммуникации с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Осуществлять коммуникации с заинтересованными сторонами. * Оценивать соответствие ИТ-продуктов предъявляемым к ним требованиям. * Подбирать графические метафоры, максимально точно соответствующие назначению разрабатываемого элемента управления. * Подготавливать графические материалы в программах подготовки векторных изображений. * Получать из открытых источников релевантную профессиональную информацию и анализировать ее. * Применять алгоритмы решения типовых задач в соответствующих областях. * Проводить интервью с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить консультации для разработчиков требований к ИТ-продуктам. * Проводить переговоры в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить переговоры с заинтересованными сторонами в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить презентации в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить презентации заинтересованным сторонам в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить ручное тестирование ИТ-продуктов. * Работать в границах заданного стиля. * Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Работать с программами прототипирования графического пользовательского интерфейса. * Работать с программами редактирования табличных данных * Работать с программами статистического анализа данных * Работать с типовой ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Разрабатывать графический дизайн интерфейсов пользователя. * Разрабатывать документы в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Разрабатывать и оформлять проектную документацию на графический пользовательский интерфейс. * Рисовать анимационные последовательности и раскадровку. * Согласовывать дизайн с заказчиком. * Создавать графические документы в программах подготовки векторных изображений. * Создавать графические документы в программах подготовки растровых изображений. * Создавать интерактивные прототипы графического пользовательского интерфейса. * Тестировать результаты прототипирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Эскизировать графические пользовательские интерфейсы. |
| **3** | **Разработка** | **54,4** |
| Специалист должен знать и понимать:   * Алгоритмы решения типовых задач, области и способы их применения. * Архитектура тестируемой системы. * Возможности настройки программного проекта в средах разработки компьютерного программного обеспечения. * Возможности типовой ИС. * Государственные стандарты испытания автоматизированных систем. * Инструменты и методы модульного тестирования. * Инструменты и методы прототипирования пользовательского интерфейса. * Инструменты и методы тестирования нефункциональных и функциональных характеристик ИС. * Интерфейсы взаимодействия внутренних модулей программного проекта. * Источники информации, необходимой для профессиональной деятельности в рамках технической поддержки процессов создания (модификации) и сопровождения ИС . * Источники информации, необходимой для профессиональной деятельности при выполнении работ по созданию (модификации) и сопровождению ИС. * Компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними. * Культура речи. * Лучшие практики создания (модификации) и сопровождения ИС в экономике. * Методологии и технологии проектирования и использования баз данных. * Методологии разработки компьютерного программного обеспечения. * Методы верификации программного обеспечения. * Методы и приемы алгоритмизации поставленных задач * Методы и приемы отладки программного кода. * Методы и приемы формализации поставленных задач * Методы и средства проверки работоспособности компьютерного программного обеспечения. * Методы и средства проверки работоспособности программных проектов. * Методы и средства сборки модулей в программный проект в средах разработки компьютерного программного обеспечения. * Методы обработки первичной информации/источников первичной информации. * методы проектирования механизированной и автоматизированной обработки информации, средства вычислительной техники, сбора, передачи и обработки информации и правила их эксплуатации, технологию механизированной и автоматизированной обработки информации, рабочие программы, инструкции, макеты и другие руководящие материалы, определяющие последовательность и технику выполнения расчетных операций, виды технических носителей информации, правила их хранения и эксплуатации, действующие системы счислений, шифров и кодов, основные формализованные языки программирования, основы программирования, методы проведения расчетов и вычислительных работ, методы расчета выполненных работ, основы экономики, организации труда и производства, правила и нормы охраны труда. * Методы работы с базами данных. * Нормативно-технические материалы по вопросам испытания и тестирования ПО. * Нотации и программное обеспечение для графического отображения алгоритмов * Область применения инструментальных средств для выполнения тестирования ПО. * Основная терминология по тестированию ПО * Основные актуальные средства генерации тестовых данных и области их применения. * Основные термины и сокращения, используемые в технической документации и принятые в организации. * Основы архитектуры мультиарендного (multitenancy) программного обеспечения. * Основы бухгалтерского учета и отчетности организаций. * Основы ИБ организации. * Основы международных стандартов финансовой отчетности. * Основы налогового законодательства Российской Федерации. * Основы организации производства. * Основы программирования. * Основы работы в операционной системе, в которой производится тестирование, на уровне, необходимом для тестирования ПО соответствующего типа. * Основы современных операционных систем. * Основы современных СУБД. * Основы теории алгоритмов и дискретной математики в объеме полученного профессионального образования. * Основы управления изменениями в проектах в области информационных технологий. * Основы управления персоналом, включая вопросы оплаты труда. * Основы управления торговлей, поставками и запасами. * Основы управленческого учета. * Особенности выбранной среды программирования и системы управления базами данных. * Особенности основных операционных систем. * Отраслевая нормативно-техническая документация. * Правила деловой переписки. * Предметная область автоматизации. * Принципы регрессионного тестирования ПО. * Программные средства и платформы инфраструктуры информационных технологий организаций. * Процедуры обеспечения безопасности при выполнении тестирования ПО. * Руководящие документы по стандартизации требований к документам автоматизированных систем. * Руководящие и нормативные материалы, регламентирующие методы разработки алгоритмов и программ и использования вычислительной техники при обработке информации; основные принципы структурного программирования; виды программного обеспечения; технико-эксплуатационные характеристики, конструктивные особенности, назначение и режимы работы ЭВМ, правила ее технической эксплуатации; технологию автоматической обработки информации; виды технических носителей информации; методы классификации и кодирования информации; формализованные языки программирования; действующие стандарты, системы счислений, шифров и кодов; порядок оформления технической документации; передовой отечественный и зарубежный опыт программирования и использования вычислительной техники; основы экономики, организации производства, труда и управления; основы трудового законодательства; правила и нормы охраны труда. * Синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования. * Синтаксис языка программирования тестируемого ПО, особенности программирования на этом языке, стандартные библиотеки языка программирования. * Системы классификации и кодирования информации, в том числе присвоения кодов документам и элементам справочников. * Системы хранения и анализа баз данных. * Современные компиляторы, отладчики и оптимизаторы программного кода. * Современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования, инструменты и методы тестирования нефункциональных и функциональных характеристик ИС. * Современные объектно-ориентированные языки программирования. * Современные подходы и стандарты автоматизации организации (CRM, MRP, ERP, …, ITIL, ITSM). * Современные стандарты информационного взаимодействия систем. * Современные структурные языки программирования. * Сообщения о состоянии аппаратных средств. * Способы использования технологических журналов, форматы и типы записей журналов. * Среды проверки работоспособности и отладки компьютерного программного обеспечения. * Теория баз данных. * Тестирование ПО, базирующееся на надежности инженерного процесса. * Тестирование ПО, ориентированное на дефекты. * Техники тестирования ПО, базирующиеся на интуиции и опыте инженера. * Техники тестирования ПО, базирующиеся на природе приложения. * Техники тестирования ПО, базирующиеся на спецификации. * Техники тестирования ПО, базирующиеся на условиях использования. * Техники тестирования ПО, ориентированные на код. * Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии. * Технологии подготовки и проведения презентаций. * Технологии программирования. * Типы и форматы сообщений об ошибках, предупреждений. * Требования по обеспечению безопасности аппаратных и программных средств автоматизированных систем, используемых при выполнении тестовых процедур, включая вопросы антивирусной защиты. * Устройство и функционирование современных ИС. * Форматы представления тестовых данных. * Язык, на котором написана техническая документация тестируемого ПО, на уровне, достаточном для чтения технической документации * Языки программирования и работы с базами данных. * Языки современных бизнес-приложений. * Языки формализации функциональных спецификаций * Языки, утилиты и среды программирования, средства пакетного выполнения процедур. |  |
| Специалист должен уметь:   * Анализировать значения полученных характеристик компьютерного программного обеспечения. * Взаимодействовать с командой разработчиков при восстановлении системы после сбоя. * Выбирать необходимые генераторы тестовых данных. * Выполнять базовую настройку операционных систем. * Выполнять модульные тесты с использованием инструментов тестирования, в том числе автоматизированного тестирования. * Выполнять процедуры сборки однородных (одноязыковых) программных модулей в программный проект в средах разработки компьютерного программного обеспечения. * Выявлять недостающую информацию для выполнения тестирования ПО в заданном объеме. * Выявлять ошибки в программном коде. * Документировать произведенные действия, выявленные проблемы и способы их устранения. * Документировать результаты проверки работоспособности компьютерного программного обеспечения. * Интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения. * Интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов. * Использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода. * Использовать выбранную среду программирования и средства системы управления базами данных. * Использовать методы и приемы алгоритмизации поставленных задач. * Использовать методы и приемы формализации поставленных задач. * Использовать программное обеспечение для графического отображения алгоритмов. * Использовать системы автоматизированного тестирования ПО. * Использовать системы контроля дефектов ПО. * Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Кодировать на языках программирования ИС в рамках технической поддержки процессов создания (модификации) и сопровождения ИС. * Кодировать на языках программирования ИС. * Находить и использовать информацию, необходимую для восстановления тестов после сбоя. * Осуществлять коммуникации с заинтересованными сторонами. * Подготавливать необходимые средства и ресурсы для выполнения задания по тестированию ПО. * Получать и использовать информацию, необходимую для выполнения задания на тестирование ПО. * Применять алгоритмы решения типовых задач в соответствующих областях. * Применять выбранные языки программирования для написания программного кода. * Применять генераторы тестовых данных. * Применять методы и приемы отладки программного кода. * Применять методы и средства проверки работоспособности компьютерного программного обеспечения. * Применять современные компиляторы, отладчики и оптимизаторы программного кода. * Применять языки программирования для написания программного кода. * Проверять на корректность отдельные модули кода ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить переговоры в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить презентации в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить проверку работоспособности программного проекта. * Производить настройки параметров программного проекта и осуществлять запуск процедур сборки. * Работать в команде со специалистами по тестированию ПО и разработчиками. * Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Работать с инструментами подготовки тестовых данных. * Работать с типовой ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Работать с типовой ИС в рамках технической поддержки процессов создания (модификации) и сопровождения ИС. * Создавать резервные копии программного проекта и данных, выполнять восстановление, обеспечивать целостность программного проекта и данных. * Составлять отчет о восстановлении работоспособности ПО. * Составлять отчет о выполнении тестирования ПО. * Составлять отчет о подготовке тестовых данных. * Составлять отчет о результатах подготовки к выполнению тестирования ПО. * Тестировать результаты кодирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Тестировать результаты прототипирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Тестировать результаты разработки ИС в рамках технической поддержки процессов создания (модификации) и сопровождения ИС. * Тестировать результаты разработки ИС. * Устанавливать корректную последовательность операций при выполнении тестирования ПО. * Устанавливать операционные системы. |
| **4** | **Документирование** | **4,4** |
| Специалист должен знать и понимать:   * Виды программного обеспечения. * Виды технических носителей информации, правила их хранения и эксплуатации. * Виды технических носителей информации. * Выделение классов эквивалентности значений каждого типа входных данных. * Выявление приоритетных областей покрытия тестовыми случаями на основе плана тестирования ПО. * Государственные стандарты испытания автоматизированных систем. * Действующие системы счислений, шифров и кодов. * Действующие стандарты, системы счислений, шифров и кодов. * Идентификация всех значений, которые вводятся конкурсантами в сценарии использования системы. * Инструктирование специалистов по подготовке тестовых данных и выполнению тестовых процедур ПО. * Инструменты и методы модульного тестирования. * Информационно-справочный и информационно-поисковый аппарат документа. * Источники информации, необходимой для профессиональной деятельности в рамках технической поддержки процессов создания (модификации) и сопровождения ИС . * Культура речи. * Лучшие практики создания (модификации) и сопровождения ИС в экономике. * Методика и стиль изложения документации пользователя (технических средств, программных средств). * Методы автоматической и автоматизированной проверки работоспособности компьютерного программного обеспечения. * Методы и приемы отладки программного кода. * Методы и средства проверки работоспособности компьютерного программного обеспечения. * Методы и средства рефакторинга, оптимизации и инспекции программного кода. * Методы классификации и кодирования информации. * Методы обработки первичной информации/источников первичной информации. * Методы оценки качества продукции в области информационных технологий. * Методы проведения расчетов и вычислительных работ. * Методы проектирования механизированной и автоматизированной обработки информации. * Методы работы с базами данных. * Методы расчета выполненных работ. * Методы тестирования ИТ-продуктов. * Написание/настройка программ для автоматизированного тестирования ПО (при необходимости). * Нормативно-правовая база применения стандартов. * Нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения изменений в системе управления версиями, порядок отражения результатов рефакторинга, оптимизации и инспекции в коллективной базе знаний. * Нормативно-технические материалы по вопросам испытания и тестирования ПО. * Область применения инструментальных средств для выполнения тестирования ПО. * Общие требования к структуре разделов технического документа. * Описание тестовых случаев. * Основная терминология по тестированию ПО * Основные актуальные средства генерации тестовых данных и области их применения. * Основные виды авторской разметки текста технической документации. * Основные виды диагностических данных проверки работоспособности компьютерного программного обеспечения и способы их представления. * Основные возможности современных текстовых процессоров. * Основные графические форматы и их особенности. * Основные методы измерения и оценки характеристик компьютерного программного обеспечения. * Основные принципы структурного программирования. * Основные способы работы с векторной и растровой графикой, способы включения рисунков в документ, правила оформления рисунков. * Основные стандарты оформления текстовых документов. * Основные стандарты оформления технической документации на компьютерное программное обеспечение. * Основные стандарты оформления технической документации. * Основные стандарты эксплуатационной документации, в том числе документации пользователя. * Основные термины и сокращения, используемые в технической документации и принятые в организации. * Основные типы эксплуатационных документов, адресованных пользователям, их особенности. * Основные формализованные языки программирования. * Основные форматы электронных документов и особенности их использования. * Основы архитектуры мультиарендного (multitenancy) программного обеспечения. * Основы бухгалтерского учета и отчетности организаций. * Основы ИБ организации. * Основы международных стандартов финансовой отчетности. * Основы налогового законодательства Российской Федерации. * Основы организации производства. * Основы программирования. * Основы работы в операционной системе, в которой производится тестирование, на уровне, необходимом для тестирования ПО соответствующего типа. * Основы современных операционных систем. * Основы современных СУБД. * Основы теории алгоритмов и дискретной математики в объеме полученного профессионального образования. * Основы типографики. * Основы трудового законодательства. * Основы управления персоналом, включая вопросы оплаты труда. * Основы управления торговлей, поставками и запасами. * Основы управленческого учета. * Основы экономики, организации производства, труда и управления. * Основы экономики, организации труда и производства, правила и нормы охраны труда. * Особенности основных операционных систем. * Передовой отечественный и зарубежный опыт программирования и использования вычислительной техники. * Перечень лидирующих инструментальных средств, их назначение, основные функциональные возможности, сильные и слабые стороны, способы применения: средства для набора текста (текстовый процессор, XML-редактор), средства подготовки снимков экрана, средства преобразования документов в выходные форматы, тестовый стенд. * Понятия «техническое средство», «программное средство», «комплекс», «система», содержание этих понятий, различия между ними. * Порядок оформления технической документации. * Порядок проектирования, производства, поставки и внедрения, применения, эксплуатации, утилизации документируемой продукции. * Построение тестовых случаев, в которых сочетаются одна перестановка значений с необходимыми внешними ограничениями. * Правила деловой переписки. * Правила и нормы охраны труда. * Предметная область автоматизации. * Процедуры обеспечения безопасности при выполнении тестирования ПО. * Рабочие программы, инструкции, макеты и другие руководящие материалы, определяющие последовательность и технику выполнения расчетных операций. * Разработка автоматизированных тестов, в том числе для проверки информационной безопасности разрабатываемого ПО. * Разработка рабочих заданий по подготовке тестовых данных и выполнению тестовых процедур ПО. * Руководящие документы по стандартизации требований к документам автоматизированных систем. * Руководящие и нормативные материалы, регламентирующие методы разработки алгоритмов и программ и использования вычислительной техники при обработке информации. * Синтаксис языка программирования тестируемого ПО, особенности программирования на этом языке, стандартные библиотеки языка программирования. * Системы хранения и анализа баз данных. * Современное состояние индустрии информационных технологий, основные подходы и тенденции. * Современные компиляторы, отладчики и оптимизаторы программного кода. * Современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования . * Сообщения о состоянии аппаратных средств. * Составление списка комбинаций значений из различных классов эквивалентности. * Способы использования технологических журналов, форматы и типы записей журналов. * Средства вычислительной техники, сбора, передачи и обработки информации и правила их эксплуатации. * Среды проверки работоспособности и отладки компьютерного программного обеспечения. * Стандарты документирования промышленной продукции, программных средств, систем (в том числе автоматизированных). * Теория баз данных. * Терминология, применяемая для описания интерфейса пользователя компьютерных систем. * Технико-эксплуатационные характеристики, конструктивные особенности, назначение и режимы работы ЭВМ, правила ее технической эксплуатации. * Технологию автоматической обработки информации. * Технологию механизированной и автоматизированной обработки информации. * Типовые метрики компьютерного программного обеспечения. * Типы и форматы сообщений об ошибках, предупреждений. * Требования по обеспечению безопасности аппаратных и программных средств автоматизированных систем, используемых при выполнении тестовых процедур, включая вопросы антивирусной защиты. * Устройство и функционирование современных ИС. * Формализованные языки программирования. * Форматы представления тестовых данных. * Человеко-ориентированный подход к проектированию интерактивных систем. * Язык, на котором написана техническая документация тестируемого ПО, на уровне, достаточном для чтения технической документации * Языки программирования и работы с базами данных. * Языки программирования и среды разработки. * Языки, утилиты и среды программирования и средства пакетного выполнения процедур. |  |
| Специалист должен уметь:   * Анализировать значения полученных характеристик компьютерного программного обеспечения. * Анализировать тестовые случаи на предмет полноты учета покрытия. * Взаимодействовать с членами команды разработчиков ПО. * Выбирать необходимые генераторы тестовых данных. * Выполнять базовую настройку операционных систем. * Выполнять модульные тесты с использованием инструментов тестирования, в том числе автоматизированного тестирования. * Выявлять недостающую информацию для выполнения тестирования ПО в заданном объеме. * Выявлять ошибки в программном коде. * Документировать результаты проверки работоспособности компьютерного программного обеспечения. * Документировать тесты в соответствии с требованиями организации. * Интерпретировать диагностические данные проверки работоспособности компьютерного программного обеспечения. * Интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов. * Использовать выбранную среду программирования для разработки процедур проверки работоспособности компьютерного программного обеспечения. * Использовать систему управления версиями для регистрации произведенных изменений. * Использовать системы контроля дефектов ПО. * Исследовать программные средства на тестовом стенде. * Исследовать продукт или технологию на тестовом стенде. * Исследовать техническую документацию, извлекать из нее сведения, необходимые для решения поставленной задачи. * Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Кодировать на языках программирования ИС. * Компоновать технический документ на основе заданных источников. * Опрашивать экспертов и анализировать полученные сведения. * Осуществлять коммуникации с заинтересованными сторонами. * Оформлять рисунки, в том числе снимки экрана, оформлять подписи к ним в соответствии с используемым стандартом. * Оформлять тестовые случаи. * Оценивать количество рабочих часов, необходимых для выполнения полученного задания. * Оценивать количество рабочих часов, необходимых для выполнения полученного задания. * Оценивать соответствие ИТ-продуктов предъявляемым к ним требованиям. * Писать программный код процедур проверки работоспособности компьютерного программного обеспечения на выбранном языке программирования. * Подготавливать графические схемы. * Подготавливать необходимые средства и ресурсы для выполнения задания по тестированию ПО. * Получать замечания у экспертов и вносить исправления в документ. * Получать и использовать информацию, необходимую для выполнения задания на тестирование ПО. * Преобразовывать технический документ в различные выходные форматы (PDF, HTML, формат электронной справки). * Применять генераторы тестовых данных. * Применять заданные стандарты и шаблоны для составления и оформления технической документации. * Применять инструментальные средства коллективной работы над программным кодом. * Применять методы и приемы отладки программного кода. * Применять методы и средства проверки работоспособности компьютерного программного обеспечения. * Применять методы, средства рефакторинга, оптимизации и инспекции программного кода. * Применять различные техники проектирования тестов (тест-дизайна). * Применять современные компиляторы, отладчики и оптимизаторы программного кода. * Применять специализированное ПО для создания автотестов. * Применять стандарты оформления кода. * Применять требования используемых в проекте стандартов с учетом особенностей данного проекта. * Применять универсальные языки моделирования (сценариев). * Применять языки программирования для написания программного кода. * Проверять на корректность отдельные модули кода ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Проводить консультации для разработчиков требований к ИТ-продуктам. * Проводить ручное тестирование ИТ-продуктов. * Публиковать результаты рефакторинга, оптимизации и инспекции в коллективной базе знаний. * Работать в команде со специалистами по тестированию ПО и разработчиками. * Работать в современном текстовом процессоре. * Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) при выполнении технической поддержки процессов создания (модификации) и сопровождения ИС. * Работать с инструментами подготовки тестовых данных. * Работать с макетами интерфейса пользователя программного средства на уровне текста, работать с ресурсными строками интерфейса пользователя программного средства. * Разрабатывать руководство по установке прикладного программного средства . * Разрабатывать руководство по эксплуатации бытового прибора. * Разрабатывать руководство пользователя прикладного программного средства. * Разрабатывать скрипты и/или программные модули для автоматизации тестирования ПО, в том числе для проверки информационной безопасности разрабатываемого ПО. * Разрабатывать технологическую инструкцию для персонала автоматизированной системы. * Разрабатывать учебное пособие по прикладному программному средству. * Современные методики тестирования разрабатываемых ИС: инструменты и методы модульного тестирования . * Создавать в тексте якоря и гипертекстовые ссылки, оформлять подписи к гипертекстовым ссылкам. * Создавать графические схемы, получать снимки экрана, включать рисунки в технический документ и оформлять их * Создавать информационно-поисковый аппарат документа с помощью текстового процессора. * Создавать, настраивать, применять стили в документе с помощью текстового процессора. * Составление списка комбинаций значений из различных классов эквивалентности. * Составлять календарный план выполнения полученного задания. * Составлять отчет о выполнении тестирования ПО. * Составлять отчет о подготовке тестовых данных. * Составлять отчет о результатах подготовки к выполнению тестирования ПО. * Составлять текст, адаптированный для автоматизированного перевода. * Составлять требования к эксплуатационному документу. * Тестировать результаты кодирования ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС. * Тестировать результаты разработки ИС. * Устанавливать корректную последовательность операций при выполнении тестирования ПО. * Устанавливать операционные системы. |
| **5** | **Внедрение** | **3** |
| Специалист должен знать и понимать:   * Виды программного обеспечения. * Виды технических носителей информации. * Действующие стандарты, системы счислений, шифров и кодов. * Информационно-справочный и информационно-поисковый аппарат документа. * Методы автоматической и автоматизированной проверки работоспособности компьютерного программного обеспечения. * Методы классификации и кодирования информации. * Методы оценки качества продукции в области информационных технологий. * Методы тестирования ИТ-продуктов. * Нормативно-правовая база применения стандартов. * Основные виды диагностических данных проверки работоспособности компьютерного программного обеспечения и способы их представления. * Основные возможности современных текстовых процессоров. * Основные графические форматы и их особенности. * Основные методы измерения и оценки характеристик компьютерного программного обеспечения. * Основные принципы структурного программирования. * Основные способы работы с векторной и растровой графикой, способы включения рисунков в документ, правила оформления рисунков. * Основные стандарты оформления текстовых документов. * Основные стандарты оформления технической документации на компьютерное программное обеспечение. * Основы типографики. * Основы трудового законодательства. * Основы экономики, организации производства, труда и управления. * Передовой отечественный и зарубежный опыт программирования и использования вычислительной техники. * Перечень лидирующих инструментальных средств, их назначение, основные функциональные возможности, сильные и слабые стороны, способы применения: текстовые процессоры, средства подготовки графических схем. * Понятия «техническое средство», «программное средство», «комплекс», «система», содержание этих понятий, различия между ними. * Порядок оформления технической документации. * Порядок проектирования, производства, поставки и внедрения, применения, эксплуатации, утилизации документируемой продукции. * Правила и нормы охраны труда. * Руководящие документы по стандартизации требований к документам автоматизированных систем. * Руководящие и нормативные материалы, регламентирующие методы разработки алгоритмов и программ и использования вычислительной техники при обработке информации. * Современное состояние индустрии информационных технологий, основные подходы и тенденции. * Стандарты документирования промышленной продукции, программных средств, систем (в том числе автоматизированных). * Технико-эксплуатационные характеристики, конструктивные особенности, назначение и режимы работы ЭВМ, правила ее технической эксплуатации. * Технологию автоматической обработки информации. * Типовые метрики компьютерного программного обеспечения. * Формализованные языки программирования. * Человеко-ориентированный подход к проектированию интерактивных систем. * Языки, утилиты и среды программирования и средства пакетного выполнения процедур. |  |
| Специалист должен уметь:   * Использовать выбранную среду программирования для разработки процедур проверки работоспособности компьютерного программного обеспечения. * Исследовать техническую документацию, извлекать из нее сведения, необходимые для решения поставленной задачи. * Компоновать технический документ на основе заданных источников. * Опрашивать экспертов и анализировать полученные сведения. * Осуществлять коммуникации с заинтересованными сторонами. * Оформлять рисунки, в том числе снимки экрана, оформлять подписи к ним в соответствии с используемым стандартом. * Оценивать количество рабочих часов, необходимых для выполнения полученного задания. * Оценивать соответствие ИТ-продуктов предъявляемым к ним требованиям. * Писать программный код процедур проверки работоспособности компьютерного программного обеспечения на выбранном языке программирования. * Подготавливать графические схемы. * Применять заданные стандарты и шаблоны для составления и оформления технической документации. * Применять требования используемых в проекте стандартов с учетом особенностей данного проекта. * Проводить консультации для разработчиков требований к ИТ-продуктам. * Проводить ручное тестирование ИТ-продуктов. * Работать в современном текстовом процессоре. * Создавать в тексте якоря и гипертекстовые ссылки, оформлять подписи к гипертекстовым ссылкам. * Создавать графические схемы, получать снимки экрана, включать рисунки в технический документ и оформлять их * Создавать информационно-поисковый аппарат документа с помощью текстового процессора. * Создавать, настраивать, применять стили в документе с помощью текстового процессора. * Составлять календарный план выполнения полученного задания. |

1.3. Требования к схеме оценки

Сумма баллов, присуждаемых по каждому аспекту, должна попадать в диапазон баллов, определенных для каждого раздела компетенции, обозначенных в требованиях и указанных в таблице 2.

Таблица 2

**Матрица пересчета требований компетенции в критерии оценки в индивидуальном формате**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Критерий / Модуль** | | | | | | | | | **Итого баллов**  **за раздел Требований компетенции** |
| **Разделы**  **Требований компетенции** |  | **А** | **Б** | **В** | **Г** | **Д** | **Е** | **Ж** |
| **1** | - | - | - | 3,30 | - | - | 2,10 | **5,40** |
| **2** | 24,30 | - | - | 0,40 | - | 6,20 | 1,90 | **32,80** |
| **3** | - | 33,00 | 15,20 | - | 6,20 | - | - | **54,40** |
| **4** | 4,40 | - | - | - | - | - | - | **4,40** |
| **5** | - | - | - | - | - | - | 3 | **3,00** |
| **Итого баллов за критерий/модуль** | | **28,70** | **33,00** | **15,20** | **3,70** | **6,20** | **6,20** | **7,00** | **100,00** |

**Матрица пересчета требований компетенции в критерии оценки**

**в командном формате**

|  |  |  |  |
| --- | --- | --- | --- |
| **Критерий/Модуль** | | | **Итого баллов**  **за раздел Требований**  **компетенции** |
| **Разделы Требований компетенции** |  | **З** |
| **1** | 6,90 | **6,90** |
| **2** | 6,00 | **6,00** |
| **3** | 36,00 | **36,00** |
| **4** | 0,00 | **0,00** |
| **5** | 0,40 | **0,40** |
| **Итого баллов за критерий/модуль** | | **49,00** | **49,00** |

1.4. Спецификация оценки компетенции

Оценка Конкурсного задания будет основываться на критериях, указанных в таблице 3.

Таблица 3

**Оценка конкурсного задания**

| **Критерий** | | **Методика проверки навыков в критерии** |
| --- | --- | --- |
| **А** | Архитектура приложения | Соответствие предметов оценки эталонным критериям и экспертному мнению экспертов-наставников.  При проверке обязательно следование Методике оценки |
| **Б** | Верстка приложения |
| **В** | Клиент-серверное взаимодействие приложения |
| **Г** | Хранение информации |
| **Д** | Взаимодействие с аппаратными расширениями устройства |
| **Е** | Тестирование |
| **Ж** | Подготовка продукта |
| **З** | Командный. Разработка мобильных приложений для бизнеса |

* 1. **Конкурсное задание**

Количество конкурсных дней: 4 дня (Д1, Д2, Д3 - в индивидуальном формате, Д4 - в командном формате).

Общая продолжительность Конкурсного задания: 27 часов, в том числе выполнение заданий:

Д1-Д3 в индивидуальном формате (модули А, Б, В, Г, Д, Е, Ж) – 21 час.

Д4 в командном формате (модуль З) – 6 часов.

Вне зависимости от количества модулей, КЗ включает оценку по каждому из разделов требований компетенции.

Оценка знаний конкурсанта проводится через практическое выполнение Конкурсного задания. В дополнение могут учитываться требования работодателей для проверки теоретических знаний / оценки квалификации.

**1.5.1. Разработка/выбор конкурсного задания**

Конкурсное задание состоит из 8 модулей, включает обязательную к выполнению часть (инвариант) – 4 модуля, и вариативную часть – 4 модуля.

Из них модуль З направлен на командообразование внутри одной компетенции с распределением функциональных задач для реализации производственной задачи с соблюдением цикла производства (проекта).

Общее количество баллов конкурсного задания в индивидуальном формате составляет 100 баллов, в командном формате – 49,00 баллов.

1.5.2. Структура модулей конкурсного задания

Необходимо разработать библиотеки и мобильное приложение для смартфона, удовлетворяющее следующим требованиям:

Минимальная версия ОС, поддерживаемая приложением, должна быть: Android - 13.0, iOS - 17.0.

В качестве бэкенда будет использован SAAS решение на примере Pocketbase, с доступом, только через REST API.

Задание будет выдано в виде задач в системе управления разработкой (таск-трекер). Каждая задача будет помечена соответствующим спринтом. Конкурсант должен выполнять поочередно спринты и в свободном режиме задачи в рамках спринта.

Каждая задача должна пройти все следующие этапы на канбан доске (предусмотрено ограничение в получении баллов в случае, если задача не удовлетворяет условия ниже):

1. Новая - здесь располагаются все задачи до выполнения.
2. В процессе - здесь задача должна находится пока над ней идет работа (в работе может быть одновременно не более 2 задач).
3. Можно проверять - сюда задача попадает автоматически при корректном коммите в системе контроля версий, интегрируемой в таск-трекер. (также конкурсант может вручную переместить карточку указав коммит в котором была выполнена задача).
4. Завершена - эксперты переносят проверенные задачи в данную колонку.

В работе необходимо использовать систему контроля версий Git, который предоставляет организатор.

Необходимо строго следовать предложенному дизайну.

Вся верстка должна быть адаптивной (следует учитывать разные размеры экранов). Необходимо:

- Избегать появления большого пустого пространства;

- Следить за отсутствием искажения элементов;

- Все элементы должны полностью находится в границах и на месте, указанном в макете;

- Учитывать расстояние между элементами;

- Используйте шрифты согласно макету. Используйте масштабирование шрифта.

Дизайн предложен в Figma:

<https://www.figma.com/design/315yyo1Ji7WGyVxBxrCWTr/Matule?node-id=45-769&t=HCpsjNwI0YfOkVed-1>

**Модуль А. Архитектура приложения (инвариант)**

**Время на выполнение модуля:** 3 часа

**Задание:**

Необходимо осуществлять комментирование кода в созданных классах. Обязательны следующие комментарии:

* Описание назначения класса.
* Автор создания.
* Описание назначения вложенных элементов программного кода (все сессии) При разработке проекта приложения вам необходимо использовать архитектуру ([см.файл с описанием архитектуры](https://docs.google.com/document/d/1DHGJzhbW0XN6D1P-USjFZkrPoljXKINbpMohblCme8I/edit?usp=sharing) ), в которой будут разделены слои бизнес-логики, представлений и домена. Изменение бизнес-логики и/или представления одного из экранов не должно повлечь за собой изменение других экранов и нарушение работоспособности приложения, за исключением переходов.Минимальное разделение на DOMAIN, PRESENTATION, DATA.

Вам предстоит разработать UIKitи сетевой слой в отдельных библиотеках (модулях).

Все вызовы методов сетевого слоя должны вызываться через уровень абстракции, так как библиотека сетевого слоя может быть заменена на другую.

Файлы проекта распределены по папкам в соответствии с архитектурой. Допустимо использование папки Common для общих файлов. Файлы настроек проекта типа info.plist, manifest.xml и прочее, допустимо оставить в папках по умолчанию.

**Модуль Б. Верстка приложения (инвариант)**

**Время на выполнение модуля:** 8 часов

**Задание:**

Вся верстка должна быть адаптивной (следует учитывать разные размеры экранов). Необходимо:

- Избегать появления большого пустого пространства;

- Следить за отсутствием искажения элементов;

- Все элементы должны полностью находится в границах и на месте, указанном в макете;

- Учитывать расстояние между элементами;

- Используйте шрифты согласно макету. Используйте масштабирование шрифта.

Дизайн предложен в Figma:

<https://www.figma.com/design/315yyo1Ji7WGyVxBxrCWTr/Matule?node-id=45-769&t=HCpsjNwI0YfOkVed-1>

1. Создайте библиотеку UIKitв виде отдельного модуля. К этому модулю необходимо создать проект где будут представлены визуально и указаны состояния компонентов в виде Storybook. (для ознакомления <https://habr.com/ru/companies/simbirsoft/articles/729066/>)

А) Создайте цвета, как на макете.

Б) Создайте тему с типографикой, как на макете.

В) Создайте шрифт из представленных иконок и интегрируйте в библиотеку.

Г) Создайте поле для ввода и все его состояния, как на макете.

Д) Создайте компонент селект, при тапе вызывается боттомшит, как на макете.

Е) Создайте компонент Поиск, как на макете.

Ж) Создайте компоненты кнопок со всеми состояниями, как на макете.

З) Создайте компонент BottomTabBar, как на макете.

И) Создайте компонент карточки, как на макете.

К) Создайте компонент «Хэдер», как на макете.

2. Создайте приложение и интегрируйте в него библиотеку UIKit. Возможно снижение баллов при нарушении принципов DRY.

1. Создан SplashScreen, как на макете. Пример анимации будет представлен в виде видео файла, который необходимо будет реализовать в приложении.

2. Создан экран авторизации, как на макете

* 1. Реализуйте проверку email на корректность (соответствие паттерну «name@domenname.ru», где имя и доменное имя может состоять только из маленьких букв и цифр). При некорректном заполнении необходимо отобразить ошибку
  2. Реализуйте возможность отображения пароля
  3. Если пользователь не найден необходимо переводить на экран Создания профиля
  4. Реализуйте обязательные требования к надежному паролю

Пароль должен содержать не менее 8 символов.

Пароль должен содержать заглавные и строчные буквы, цифры, пробелы и специальные символы.

Например: oNQZnz$Hx2.При нажатии на кнопку «Sign Up» осуществляется переход на экран «Log In»

3. Создан экран Создания профиля, как на макете

* 1. Все поля обязательны для ввода

4. Создан экран Создания пароля для открытия приложения, как на макете

5. Создан экран Создания пароля OTP, как на макете

6. Создан экран Создания пароля для авторизации, как на макете

* 1. Реализуйте возможность отображения пароля

7. Создан экран Главная, как на макете

* 1. При выборе категории описания фильтруются

8. Создан экран Каталога, как на макете

* 1. Реализуйте поиск
  2. Реализуйте добавление в корзину
  3. Реализуйте удаление из корзины

9. Создан экран Корзины

* 1. Реализуйте изменение корзины

10. Создан боттомшит карточки описания

11. Создан экран Список проектов

12. Создан экран Создания проекта

* 1. Все поля обязательны для ввода
  2. При добавлении фото присутствует предпросмотр

13. Создан экран Просмотра проекта

14. Создан экран Профиля

**Модуль В. Клиент-серверное взаимодействие приложения (инвариант)**

**Время на выполнение модуля:** 3 часа

**Задание:**

Необходимо создать библиотеку, которая будет содержать следующие элементы:

А) Модели приходящие от сервера

Б) Сетевой клиент

В) Запросы к серверу

Г) Обработки ошибок

Необходимо корректно обрабатывать запросы к серверу. В случае отсутствия соединения с сетью Интернет необходимо отобразить экран как на макете. В случае получения ошибки от сервера или отсутствия соединения с сетью Интернет необходимо отобразить соответствующий текст ошибки в диалоговом окне, которое должно закрываться только пользователем.

Запросы к серверу:

1) Авторизация

2) Создание профиля и ОТР

3) Получение акций

4) Получение каталога

5) Поиск

6) Получение описания

7) Добавление в корзину

8) Изменение корзины

9) Оформление заказа

10) Список проектов

11) Создание проекта

12) Просмотр проекта

13) Получение информации о профиле

14) Выход

1. Экран авторизации, взаимодействует с сервером через абстракцию и библиотеку

2. Экран Создания пароля OTP, взаимодействует с сервером через абстракцию и библиотеку

3. Экран пароля для авторизации, взаимодействует с сервером через абстракцию и библиотеку

4. Экран Главная, взаимодействует с сервером через абстракцию и библиотеку

5. Экран Каталога, взаимодействует с сервером через абстракцию и библиотеку

6. Экран Корзины, взаимодействует с сервером через абстракцию и библиотеку

7. Экран Список проектов, взаимодействует с сервером через абстракцию и библиотеку

8. Экран Создания проекта, взаимодействует с сервером через абстракцию и библиотеку

9. Экран Просмотра проекта, взаимодействует с сервером через абстракцию и библиотеку

10. Экран Профиля, взаимодействует с сервером через абстракцию и библиотеку

**Модуль Г. Хранение информации (инвариант)**

**Время на выполнение модуля:** 1 час

**Задание:**

Все медиа ресурсы должны кэшироваться.

- Создана ветка для спринта

- Проект корректно сохранен в ветку Спринт Х и не требует дополнительного разархивирования.

- Выполнен pullrequest ветки Спринта Х в ветку main и закрыт

1. Реализуйте возможность ознакомления с Политикой конфиденциальности (PDF)
2. Реализуйте возможность ознакомления с Пользовательским соглашением
3. Реализуйте кэширование главной страницы
   1. Данные входа и активная сессия должны быть сохранены и после перезапуска приложения
4. Необходимо сохранять состояние экрана Создание профиля после перезапуска приложения.
5. Необходимо сохранять состояние после перезапуска приложения

**Модуль Д. Взаимодействие с аппаратными расширениями устройства (инвариант)**

**Время на выполнение модуля:** 2 часа

**Задание:**

1. Экран Создания проекта:
   1. При нажатии на “Выбрать фото” необходимо пользователю предоставить возможность выбрать фотографию из галереи или сделать фото с камеры устройства.
2. Создай локальных уведомления при не активности в последнем открытом проекте более 3 минут
3. Приложение должно поддерживать смену темы на основе темы в системе
4. Необходимо, чтобы приложение отправляло локальное уведомление, если пользователь не открывал приложение более минуты. Уведомление должно отправляться один раз до следующего входа в приложение.
5. Виджет:
   1. Необходимо создать виджет для домашнего экрана операционной системы счетчика рядов в избранном проекте. Если пользователя не прошел авторизацию в приложении, то необходимо отображать призыв к действию

**Модуль Е. Тестирование (вариатив)**

**Время на выполнение модуля:** 3 часа

**Задание:**

* + - 1. Создайте тесты состояний в библиотеки UIKitк следующим компонентам:

1. Поле для ввода
2. Селект
3. Кнопки
4. BottomTabBar
5. Карточка
   * + 1. Создайте тесты на библиотеку сетевого слоя на каждый из запросов, по тест кейсам.

**Запросы к серверу**:

1) Авторизация

2) Создание профиля и ОТР

3) Получение акций

4) Получение каталога

5) Поиск

6) Получение описания

7) Добавление в корзину

8) Изменение корзины

9) Оформление заказа

10) Список проектов

11) Создание проекта

12) Просмотр проекта

13) Получение информации о профиле

14) Выход

**Модуль Ж. Подготовка продукта (вариатив)**

**Время на выполнение модуля:** 1 час

**Задание:**

Необходимо во время сессии работать в ветке “Path-X”, где Х – это номер сессии. По завершению сессии необходимо сделать средствами GiltabMerge, с основной веткой, которая должна называться “main”, при этом ветка удалятся не должна.

1. Создайте презентацию:
   * Презентация должна быть рассчитана на разработчиков (Вы не должны продавать приложение!).
   * В презентации нужно рассказать о реализации приложения.
   * Показать схему классов.
   * Производительность.
   * Используемые архитектурные решения.
   * Используемые библиотеки или описание почему не использовались.
2. Подготовьте приложение к публикации в одном из магазинов предложенном главным экспертом. (напримерruStore) Следующие действия необходимо делать в магазине, предоставив экспертам доступ к приложению (Добавить разработчика - учетная запись экспертов или войти для проверки на рабочем месте эксперта):
   * Создайте приложение с названием, первая буква фамилии латиницей + matule.
   * Приложение должно быть бесплатным.
   * Загрузите иконку приложения для магазина.
   * Добавьте “Название для пользователя”.
   * Добавьте Тип .
   * Выберете Категорию.
   * Укажите возрастную категорию.
   * Добавьте краткое описание.
   * Добавьте основное описание.
   * Добавьте Скриншоты приложения (необходимо загрузить все размеры необходимые для публикации на смартфон.

**Модуль Ж. Командный. Разработка мобильных приложений для бизнеса (вариатив)**

**Время на выполнение модуля:** 6 часов

**Задание:**

Необходимо разработать кроссплатформенное мобильное приложение для смартфонов:

* Android: минимальная версия 8.0 (API 26).
* iOS: минимальная версия iOS 13.

**Ориентация:** поддержка только портретной ориентаций экрана. В случае, если на устройстве активирована функция поворота экрана, то при повороте устройства поворота экрана не происходит.

**Требования к локализации:**

* Язык интерфейса приложения: Русский язык.
* Единицы измерения: Метрическая система.
* Время и дата: формат “01.01.2025”.
* Часовые пояса: используется часовой пояс с устройства пользователя. Все даты передаются в UTC.

**Описание проекта**

Корпоративное приложение предназначено для HR-специалистов компании и выполняет функции по учету соискателей, ведения истории событий взаимодействия с ними, а также для учета открытых/закрытых в компании вакансий.

**Авторизация**

**API:**

1. **post/auth**

Метод аутентификации пользователя.

1. **get /auth/profileByToken**

Метод получения данных профиля пользователя по токену доступа.

**Элементы интерфейса и логика работы**

**Элементы интерфейса:**

1. Логотип
2. Поля для ввода:

**Элементы Логина:**

* Обязательное текстовое поле.

**Элементы Пароля:**

* Обязательное текстовое поле.
* Возможность скрыть/показать пароль.

1. Чекбокс "Запомнить":

* По умолчанию чекбокс не отмечен.
* При выходе из приложения (при завершении сеанса) текущая сессия должна удаляться.

1. Кнопка "Войти".
2. Ссылка "Забыли пароль?".

**Логика работы:**

1. HR-специалист вводит логин и пароль, нажимает кнопку «Войти».
2. При наличии ошибок валидации, при попытке нажатия на кнопку «Войти» отображается сообщение об ошибке под соответствующим полем, текст подсвечивается красным
3. По нажатию кнопки "Войти" отправляется запрос на сервер с данными логина и пароля
4. Если запрос успешен, переходим на следующий экран: экран загрузки в виде анимации бегущей полосы (слева направо).
5. После экрана загрузки – переход на экран HR-board

**4.1. Ошибки запроса:**

Если введены некорректные логин или пароль, отображается сообщение об ошибке под соответствующим полем, текст подсвечивается красным. Если пользователь заблокирован, то пользователь не может войти, учетка может быть восстановлена только с помощью администратора. Также, необходимо выводить сообщение, если сервер недоступен

**1.2. Сценарий нажатия ссылки «Забыли пароль?»**

1.2.1.По нажатию на ссылку «Забыли пароль?» происходит переход на веб-страницу восстановления пароля.

**Bottom bar, logout**

**API:**

1. **get /logout**

Метод выхода из ЛК.

**Элементы интерфейса и логика работы**

**1 – Кнопка «Выйти из ЛК».**

Логика работы: при нажатии HR-специалистом на данную кнопку появляется диалоговое окно с подтверждением выхода из ЛК и после подтверждения осуществляется переход на экран авторизации

Токен авторизации должен удаляться из локального хранилища.

При нажатии на кнопку «Отмена» происходит возврат на соответствующую вкладку, с которой HR-специалист нажимал на кнопку «Выйти из ЛК».

**2 – Панель вкладок.** Панель должна содержать три вкладки: «HR доска», «Соискатели», «Вакансии».

Логика работы: при нажатии на вкладку она выделяется цветом и происходит переход на соответствующий экран выбранного раздела.

**Вакансии. Главная**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **get/vacancy**

Метод получения списка вакансий.

**Элементы интерфейса и логика работы**

**Экран загрузки.**

**1 – Кнопка «Выйти из ЛК».**

**2 – Кнопка «Настроить таблицы»** - позволяет HR-специалисту смотреть необходимый набор данных, отображаемый в краткой информации о вакансии.

Логика работы: при нажатии на данную кнопку открывается экран «Настройки таблицы».

**3 – Кнопка «Создать вакансию»**: позволяет HR-специалисту создать новую вакансию. Доступно только для рекрутера.

Логика работы: при нажатии на кнопку открывается экран создания новой вакансии с формой для заполнения.

**4 - Название раздела**.

**5 – Поисковая строка** (логика описана ниже).

**6 - Кнопка «Применить фильтры»**: позволяет HR-специалисту отфильтровать список вакансий по необходимым параметрам.

Логика работы: при нажатии на данную кнопку открывается экран «Фильтры».

**7 – Анимация загрузки**

**8 – Нижняя навигационная панель**.

**1 - Информационная строка «n вакансий найдено»**: отображает количество вакансий, соответствующих текущему поисковому запросу и/или примененным фильтрам.

Логика работы: при первоначальной загрузке экрана отображается общее количество вакансий (например, «25 вакансий найдено»).

При сортировке вакансий при помощи строки поиска или применении фильтров текст обновляется, отражая количество вакансий, соответствующих текущим условиям.

**2 – Список вакансий**: отображается список всех вакансий с краткой информацией.

Логика работы: при нажатии на элемент списка открывается экран с деталями вакансии.

Краткая информация изменяется при настройке таблиц. Список вакансий обновляется при изменении фильтров или поискового запроса.

Изначально выводятся только: название, должность, дата открытия, подразделение.

**Реализация поиска:**

Строка поиска: позволяет HR-специалисту искать вакансии по ключевым словам.

* Логика работы: поиск выполняется по ключевым словам, содержащимся в названии вакансии. Поиск не должен быть чувствителен к регистру. Поиск запускается после нажатия на кнопку. После успешного поиска выводится кол-во найденных вакансий в алфавитном порядке.

**Вакансии. ID**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **get/vacancy/{id}**

Метод получения вакансии по ID.

1. **delete/vacancy/{id}**

Метод удаления вакансии.

**Элементы интерфейса и логика работы**

**Просмотр детальной страницы вакансии: доступ – сорсер:**

Для сорсера доступен только просмотр информации о вакансии.

**Элементы интерфейса:**

1 – кнопка «Назад»: позволяет HR-специалисту покинуть экран просмотра детальной информации о вакансии.

Логика работы: при нажатии на кнопку происходит возврат на главный экран раздела «Вакансии».

2 – Детальная информация о вакансии.

Данные выводятся при запросе [get/vacancy/{id}](#_get/vacancy/{id}_1).

Из ответа выводятся следующие данные:

* Название вакансии;
* Подразделение. Отображается название подразделения
* Дата открытия.;
* Должность;
* Статус;
* Зарплата;
* Публичная вакансия.;
* Опыт. Отображаются следующие текстовые значения:
* «Не требуется»
* «1-3 года»
* «3-6 лет»
* «Более 6 лет».
* "Город;
* Описание / обязанности;
* Условия;
* Требования;
* Технические вопросы;
* HR-менеджер;
* Номер телефона;
* Электронная почта;
* Куратор вакансии.

**Просмотр детальной страницы вакансии: доступ – рекрутер:**

Элементы интерфейса аналогичны элементам интерфейса с доступом сорсера.

1 – Кнопка «Редактировать вакансию»: при нажатии на данную кнопку поисходит переход на экран редактирования вакансии.

2 – Кнопка «Удалить вакансию»

3 – Предупреждения перед удалением вакансии.

4 – Кнопка «Отмена»: при нажатии происходит возврат на экран Рис.3-4.

5 – Кнопка «Удалить»: при нажатии на данную кнопку происходит переход на главный экран раздела «Вакансии». Вакансия удаляется (метод [delete/vacancy/{id}](#_delete/vacancy/{id})).

Рекрутер может удалять только те вакансии, на которые не назначен ни один соискатель, и с которыми не связано ни одно событие. В случае, если рекрутер пытается удалить вакансию с назначенными соискателями и/или со связанным событием, появляется ошибка.

**Вакансии. Настройка таблиц**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **get/vacancy**

**Элементы интерфейса и логика работы**

**Элементы интерфейса (Рекрутер):**

**1 – Кнопка «Назад»:** при нажатии на эту кнопку происходит возврат на экран вакансий

**2 – Название**

**3 – Параметры вакансии**: по умолчанию отмечены «должность», «дата открытия», «подразделение». Параметр «Название» не доступен для нажатия. Носит информативный, наглядный характер. По умолчанию всегда отображается при любой комбинации параметров таблиц.

Логика работы: при нажатии на дополнительный параметр появляется галочка, т.е. параметр выбран для отображения. При повторном нажатии на выбранный параметр галочка пропадает, параметр не должен отображаться на главном экране.

Аналогично для отмеченных по умолчанию параметров: при нажатии на параметр галочка пропадает, параметр не отображается на главном экране.

**4 – Кнопка «Применить»**: при нажатии на данную кнопку изменения сохраняются. Метод: [get/vacancy](#_get/vacancy).

**5 – Кнопка «Настройки по умолчанию»**: при нажатии на данную кнопку происходит переход на экран вакансий. Отображаются только базовые параметры («название», «должность», «дата открытия», «подразделение»)

**Вакансии. Фильтры**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **get/vacancy**

Метод получения списка вакансий.

1. **get/hiring/departments**

Метод получения подразделений. Выводится поле «name».

1. **get/hiring/positions**

Метод получения списка должностей.

1. **get/vacancy/short**

Данный метод выводит краткую информацию о вакансиях.

1. **get/employee/shortStaff**

Данный метод выводит краткую информацию обо всех HR-специалистах

**Элементы интерфейса и логика работы (доступ - рекрутер)**

**1 – Кнопка «Назад»**: при нажатии происходит возврат на главный экран раздела «Вакансии».

**2 – Название**.

**3 – Параметры фильтрации**: при нажатии на параметр происходит переход на соответствующий экран выбора значения параметра.

**4 – Кнопка «Применить»**: если ни один параметр не настроен, то кнопка неактивна. При наличии выбранных параметров фильтрации, при нажатии на кнопку, после экрана загрузки на главном экране раздела «Вакансии» отображаются только те вакансии, которые удовлетворяют условиям фильтрации.

**5 – Кнопка «Сбросить все фильтры»**: при нажатии на данную кнопку появляется диалоговое окно.

При нажатии на кнопку «Сбросить» все фильтры сбрасываются

При нажатии на кнопку «Отменить» возврат на текущий экран.

**6 – Выбранные параметры фильтрации**: при нажатии на элемент крестик выбранный параметр удаляется.

Можно настраивать множество параметров одного типа, т.е. можно выбрать несколько названий, должностей и тд.

**Название**

API: [get/vacancy/short](#_get/vacancy/short).

**1 – Кнопка «Отмена»**: при нажатии на данную кнопку происходит возврат к экрану вакансий

**2 – Название**.

**3 – Кнопка «Готово»**: при нажатии на данную кнопку выбранное значение / выбранные значения сохраняются и отображаются.

**4 – Поисковая строка**: позволяет HR-специалисту искать должность по ключевым словам.

* Логика работы: поиск выполняется по ключевым словам, содержащимся в названии вакансии. Поиск не должен быть чувствителен к регистру. Поиск запускается после нажатия на кнопку. После успешного поиска выводится кол-во найденных вакансий в алфавитном порядке.

**5 – Список названий всех вакансий** в алфавитном порядке. По умолчанию отмечено значение «Не выбрано». HR-специалист может выбрать одно или несколько значений.

При выборе (при нажатии) какого-либо параметра появляется он отмечается, при повторном нажатии выделение снимается, параметр не будет применен в качестве фильтра.

Если рекрутер убрал выбор со всех элементов, то по умолчанию снова отображается отмеченный параметр «Не выбрано».

**HR-менеджер**

API: [get/employee/shortStaff](#_get/employee/shortStaff).

Для данного списка выводятся все действующие HR-специалисты компании в алфавитном порядке.

Элементы интерфейса и логика работы аналогичны п. [Название](#_Название).

**Должность**

**API:** [**get/hiring/positions**](#_get/hiring/positions)**.**

Для данного списка выводятся все названия должностей в алфавитном порядке.

Элементы интерфейса и логика работы аналогичны п. [Название](#_Название).

Первоначально список выводится методом апи и кэшируется. При дальнейших запросах список выводится из кэша.

**Дата открытия**

Элементы интерфейса:

1 – Поле отображения начальной даты: по умолчанию отображается текущая дата.

2 – Поле отображения конечной даты: по умолчанию отображается текст «мм.дд.гггг.».

Логика работы: при нажатии на данное поле HR-специалист может установить диапазон дат, выбрав конечную дату диапазона на календаре. Установленная конечная дата отображается в этом же поле.

3 – Кнопки «Назад» и «Вперед»: позволяют HR-специалисту производить поиск месяца даты открытия вакансии.

4 – Интерактивные даты. Календарь отображает текущие день, месяц и год по умолчанию.

5 – Кнопка «Отменить»: позволяет HR-специалисту отменить применение фильтра по дате.

6 – Кнопка «Применить»: позволяет HR-специалисту сохранить выбранные дату/диапазон дат.

Можно выбрать как одну конкретную дату, так и диапазон дат.

**Статус**

Элементы интерфейса и логика их работы аналогичны элементам п. [Название](#_Название).

Для данного фильтра выводятся три фиксированных значения: «Любое состояние заявки» - отмечено по умолчанию, «Только открытые заявки» и «Только закрытые заявки».

**Подразделения**

API: [get/hiring/departments](#_get/hiring/departments).

Для данного списка выводятся все названия подразделений в алфавитном порядке.

Элементы интерфейса и логика работы аналогичны п. [Название](#_Название).

Первоначально список выводится методом апи и кэшируется. При дальнейших запросах список выводится из кэша.

**Тип заявки**

Элементы интерфейса и логика их работы аналогичны элементам п. [Название](#_Название).

Для данного фильтра выводятся три фиксированных значения: «Любой тип заявки» - отмечено по умолчанию, «Только публичные заявки» и «Только не публичные заявки».

**Куратор**

API: [get/employee/shortStaff](#_get/employee/shortStaff).

Для данного списка выводятся все действующие сотрудники компании в алфавитном порядке.

Элементы интерфейса и логика работы аналогичны п. [Название](#_Название).

**Экран фильтров доступа «сорсер»**

Для сорсера элементы интерфейса и логика аналогичны. Различие заключается в составе фильтров. У сорсера их меньше: Название, Должность, Дата открытия, Подразделение, HR-менеджер, Куратор

**Вакансии. Создание вакансии**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **post/vacancy**

Метод создания вакансии.

1. [**get**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Employee_management/getByEmployeeId)**/employee/{id}**

Метод используется для получения эл. почты HR-менеджера.

1. **[post](http://smartstaff-dev.simbirsoft:8080/swagger/" \l "!/Vacancy_management/uploadFiles) /vacancy/{id}/upload-files**

Метод предназначен для сохранения прикрепленных файлов из формы создания вакансии.

1. **get/dictionary/city**

Метод получения списка городов. В Swagger метод напрямую отсутствует, есть метод dictionary/types:

**Элементы интерфейса и логика работы**

**1 – Заголовок**..

**2 – Кнопка «Закрыть»**: при нажатии на данную кнопку появляется диалоговое окно

При нажатии на кнопку «Выйти» рекрутер возвращается на главный экран раздела «Вакансии», никакие изменения не сохраняются

**3 – Progress bar**: показывает, на каком этапе заполнения формы находится рекрутер

**4 – Заголовок формы**.

**5 – Поле «Название»**. Обязательное поле.

**6 – Поле «Подразделение»**. Поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора подразделения).

**7 – Поле «Дата открытия»**. Обязательное поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора даты).

**8 – Поле «Должность»**. Обязательное поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора должности).

**9 – Поле «Зарплата»**. Обязательное поле.

**10 - Чекбокс «Публичная вакансия»**. По умолчанию не отмечен.

**11 – Поле «Требуемое количество»**. Обязательное поле.

**12 – Поле «Опыт»**. Поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора диапазона опыта).

**13 – Поле «Город»**: поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора города).

**14 – Информационное сообщение**.

**15 – Кнопка «Продолжить»**. Изначально неактивна. При заполнении всех обязательных полей становится активной, и рекрутер может перейти к следующему шагу заполнения формы.

**16 – Кнопка «Назад»**. При нажатии на данную кнопку происходит возврат к предыдущему этапу заполнения формы

**17 – Название элемента формы**.

**18 - Поле «Описание / обязанности»**. Обязательное поле.

**19 – Название элемента формы**.

**20 – Поле «Условия»**.

**21 – Название элемента формы**.

**22 – Поле «Требования»**.

**23 – Название элемента формы**.

**24 – Поле «Технические вопросы»**.

**25 – Название элемента формы**.

**26 – Поле «Тестовое задание»**.

**27 – Название элемента формы**.

**28 – Прикрепленный файл**: a – Изображение. Для всех прикрепленных файлов одинаково;

b – Название файла;

c – Дата, время загрузки файла, его размер;

d – Кнопка «Удалить прикрепленный файл». При нажатии на данную кнопку появляется диалоговое окно

**29 – Информационное сообщение** о том какие типы файлов и какой размер можно загрузить

**30 – Кнопка «Загрузить файл»**. Рекрутер может выбрать один файл из локального хранилища или галереи. После выбора файл должен быть сохранен в локальное хранилище приложения.

**31 – Информационное сообщение**. Поясняет, что поля, отмеченные «\*», обязательны для заполнения.

**32 – Кнопка «Продолжить»**. Изначально неактивна. При заполнении обязательного поля становится активной, и рекрутер может перейти к следующему шагу заполнения формы

**33 – Название блока полей**.

**34 – Поле «HR-менеджер»**. Поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора HR-менеджера. Для списка «HR-менеджер» выводятся все действующие HR компании).

**35 – Поле «Номер телефона»**.

**36 – Поле «Электронная почта»**.

* После выбора HR-менеджера сразу производится дозапрос на получение электронной почты - /rest/employee/{id}. Поле можно заполнить вручную или отредактировать полученный с сервера ответ.
* Из ответа выводим значение «email».

**37 – Название поля.**

**38 – Поле «Ответственный»**. Поле с возможностью выбора (при нажатии на поле происходит переход на экран выбора ответственного лица. Для списка ответственного лица выводятся все действующие HR компании).

**39 – Кнопка «Создать вакансию»**.

Все данные сохраняются методом [post/vacancy](#_post/vacancy). Отдельным методом сохраняются прикрепленные документы: [post /vacancy/{id}/upload-files](#_post_/vacancy/{id}/upload-files). Прикрепленные документы отображаются в просмотре вакансии

**Ошибки валидации**

Должны выводиться три типа ошибок валидации:

1. Превышение максимально допустимого количества символов
2. Ввод неверного типа данных
3. Попытка загрузки файла неподходящего формата и/или размера.

При ошибках валидации кнопки «Продолжить» и «Создать вакансию» становятся неактивными до устранения ошибки

**Ошибка 404 запроса** [**post**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Vacancy_management/uploadFiles)[**/vacancy/{id}/upload-files**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Vacancy_management/uploadFiles)

Необходимо выводить текст ошибки если файлы были потеряны при загрузке

**Ошибка сохранения**

При возникновении ошибки при сохранении вакансии появляется диалоговое окно

**Вакансии. Редактирование вакансии**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **put/vacancy/{id}**

Метод редактирования вакансии.

1. **post/vacancy/{id}/upload-files**

Метод удаления прикрепленного файла / прикрепления нового файла.

1. **[get](http://smartstaff-dev.simbirsoft:8080/swagger/" \l "!/Vacancy_management/getById)/vacancy/{id}**

Метод предназначен для получения данных конкретной вакансии.

**Элементы интерфейса и логика работы**

При нажатии на кнопку на экране дательной информации вакансии происходит переход на экран редактирования вакансии. В режиме редактирования заголовок формы (2) меняется на название редактируемой вакансии. Данные в форму подгружаются из [get/vacancy/{id}](#_get/vacancy/{id}):

Логика заполнения полей такая же, как при создании вакансии.

При нажатии на кнопку все внесенные изменения сохраняются [put/vacancy/{id}](#_put/vacancy/{id}), [post/vacancy/{id}/upload-files](#_post/vacancy/{id}/upload-files), высвечивается сообщение об успешном сохранении, отображается обновленная информация о вакансии.

При отмене редактирования без сохранения, появляется диалоговое окно.

В случае ошибки редактирования вакансии отображается диалоговое окно с ошибкой сохранения

**Соискатели. Главная**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.
* Карточка соискателя — это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При желании можно настроить отображаемую краткую информацию. При нажатии на карточку происходит переход в полный профиль соискателя, где можно ознакомиться с детальной информацией.

**API:**

1. **[post](http://smartstaff-dev.simbirsoft:8080/swagger/" \l "!/Employee_management/getApplicantsBy) /employee/applicantListBy**

Метод предназначен для вывода списка соискателей.

1. **post/vacancy/{id}/upload-files**

Метод удаления прикрепленного файла / прикрепления нового файла.

1. [**get**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Vacancy_management/getById)**/vacancy/{id}**

Метод предназначен для получения данных конкретной вакансии.

**Элементы интерфейса и логика работы**

**Экран загрузки**

1 – Кнопка «Выйти из ЛК».

2 – Кнопка «Настроить таблицы» - позволяет HR-специалисту настроить необходимый набор данных, отображаемый в карточке соискателя.

Логика работы: при нажатии на данную кнопку открывается экран «Настройки таблицы».

3 – Кнопка «Добавить нового соискателя»: позволяет HR-специалисту создать запись нового соискателя.

Логика работы: при нажатии на кнопку открывается экран добавления нового соискателя.

4 – Название раздела.

5 – Кнопка «Применить фильтры»: позволяет HR-специалисту отфильтровать список соискателей по необходимым параметрам.

Логика работы: при нажатии на данную кнопку открывается экран «Фильтры».

6 – Поисковая строка

7 – Анимация загрузки.

8 – Нижняя навигационная панель.

**Главный экран**

**1 – Информационная строка «n соискателей»**: отображает количество соискателей, соответствующих:

* общему количеству соискателей при первоначальной загрузке экрана;
* количеству соискателей при применении фильтров.
* количеству соискателей при их сортировке посредством поисковой строки. В данном случае текст меняется на: «n соискателей найдено».

Логика работы: при первоначальной загрузке экрана отображается общее количество соискателей (например, «8062 соискателей»).

При сортировке соискателей при помощи строки поиска и/или применении фильтров появляется экран загрузки, затем текст обновляется, отображая количество соискателей, соответствующих текущим условиям.

Если соискатели не найдены, отображается текст «Ничего не найдено» (Рис.6).

**2 – Список соискателей**: отображается список всех соискателей с краткой информацией

Запрос: [post /employee/applicantListBy](#_post_/employee/applicantListBy).

Через 50 соискателей отображается кнопка «Загрузить еще». При нажатии на эту кнопку делается новый запрос и далее по аналогии с шагом 50.

По умолчанию в краткой информации соискателя отображаются:

Город, ФИО, Желаемая позиция, Статус, Телефон, Фото соискателя: для каждого соискателя загружается по id:

Метод GET,

URL: /userAvatar?id=

Логика работы: при нажатии на элемент списка открывается экран профиля соискателя.

Краткая информация изменяется при настройке таблиц.

**3 – Кнопка «Копировать»**: позволяет HR-специалисту копировать номер телефона соискателя в буфер обмена.

Требуется адаптировать все вышеуказанные элементы под разные размеры экранов.

**Реализация поиска:**

Строка поиска позволяет HR-специалисту искать соискателей по краткой информации карточки.

Логика работы: поиск выполняется по ключевым словам, содержащимся в карточке соискателя, т.е. по всей краткой информации соискателя.

Если пользователь ввел текст в строку поиска, а затем полностью стер его, система должна обработать этот случай как пустой запрос. В этом случае отображаются все соискатели без фильтрации.

HR-специалист вводит текст в строку поиска. Поиск не чувствителен к регистру. После ввода необходимого запроса HR-специалист нажимает на кнопку «Готово», показываются все соискатели в алфавитном порядке.

**Соискатели. Настройка таблиц**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.
* Карточка соискателя — это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При желании можно настроить отображаемую краткую информацию. При нажатии на карточку происходит переход в полный профиль соискателя, где можно ознакомиться с детальной информацией.

**API:**

1. [**post**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Employee_management/getApplicantsBy)**/employee/applicantListBy**

Метод предназначен для вывода списка соискателей.

**Элементы интерфейса и логика работы**

Логика работы аналогична логике настройке таблицы вакансий.

**1 – Кнопка «Назад»**.

**2 – Topbar**.

**3 – Настраиваемые параметры**.

**6 – Кнопка «Настройки по умолчанию»**. При нажатии на данную кнопку происходит сброс всех настроенных параметров, выводятся только элементы по умолчанию: «Город», «ФИО», «Желаемая позиция», «Статус», «Телефон», «Фото соискателя». Фото соискателя, как и ФИО, выводится при любых настройках таблиц.

Кнопки «Применить» и «Настройки по умолчанию» закреплены в нижней части.

**Соискатели. Фильтры**

**Термины и определения**

* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.
* Карточка соискателя — это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При желании можно настроить отображаемую краткую информацию. При нажатии на карточку происходит переход в полный профиль соискателя, где можно ознакомиться с детальной информацией.

**API:**

1. [**post**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Employee_management/getApplicantsBy)**/employee/applicantListBy**

Метод предназначен для вывода списка соискателей.

1. **get/hiring/positions**

Метод получения списка должностей.

1. **get/dictionary/city**

Метод получения списка городов

1. **get/dictionary/status**

Метод получения списка статусов.

1. **get/employee/shortStaff**

Данный метод выводит краткую информацию обо всех HR-специалистах

**Элементы интерфейса и логика работы**

1

1 – Top bar: включает в себя заголовок и кнопку «Назад».

2 – Меню фильтров. Состоит из разных типов элементов:

c – Строка навигации. При нажатии на данную строку происходит переход на экран выбора параметра / параметров из списка.

d – Переключатель. Выбор между двумя значениями: «Да» / «Нет».

e – Информационное сообщение. Поясняет, что при нажатии на переключатель можно искать позиции по И.

f – Строка навигации. При нажатии на данную строку происходит переход на экран выбора даты при помощи календаря.

g – Числовые поля. Включает в себя две кнопки: «+» и «-».

3 – Кнопка «Применить»: зафиксирована в нижней части экрана, при скролле всегда остается видимой. Если ни один параметр не настроен, то кнопка неактивна.

4 – Кнопка «Сбросить все фильтры»: при нажатии на данную кнопку появляется диалоговое окно

При нажатии на кнопку «Сбросить» все фильтры сбрасываются

5 – Выбранные параметры фильтрации: при нажатии на крестик выбранный параметр удаляется.

Можно настраивать множество параметров, т.е. можно выбрать несколько позиций, городов и тд.

**Желаемая позиция**

API: [get/hiring/positions](#_get/hiring/positions_1).

**1 – Заголовок**:

**2 – Кнопка «Отмена»**: при нажатии на данную кнопку происходит возврат, выбранные параметры не сохраняются.

3 – Кнопка «Готово»: при нажатии на данную кнопку выбранное значение / выбранные значения сохраняются и отображаются

Если отмечено значение «Не выбрано», то оно не отображается в выбранных параметрах фильтрации

**4 – Поисковая строка**: позволяет рекрутеру искать позицию по ключевым словам.

Логика работы: поиск выполняется по ключевым словам, содержащимся в названии позиции.

**5 – Список всех позиций.** Позиции расположены в алфавитном порядке. Позиции должны быть актуальными.

Возможен выбор как одного элемента списка, так и нескольких.

Если включен «Поиск позиции по «И», то при выборе двух и более позиций поиск производится не по отдельным позициям, а по наличию всех выбранных позиций в карточке соискателя.

**Город**

API: [get/dictionary/city](#_get/dictionary/city_1).

**1 – Заголовок**.

**2 – Кнопка «Отмена**

**3 – Кнопка «Готово»**:

**4 – Поисковая строка**: позволяет рекрутеру искать город по ключевым словам. Поиск аналогичен п. [Желаемая позиция](#_Желаемая_позиция).

**5 – Список всех городов.** Города расположены в алфавитном порядке.

Возможен выбор как одного элемента списка, так и нескольких.

**Статус**

API: [get/dictionary/status](#_get/dictionary/status).

**1 – Заголовок**:

**2 – Кнопка «Отмена»**.

**3 – Кнопка «Готово»**

**4 – Список всех статусов**. Статусы расположены в алфавитном порядке (А-Я).

Возможен выбор как одного элемента списка, так и нескольких.

Список статусов кэшируется. После 1-го запроса выводится из кэша.

**Уровень английского**

**1 – Заголовок**.

**2 – Кнопка «Отмена»**.

**3 – Кнопка «Готово****»**

**4 – Список всех уровней английского**.

Возможен выбор как одного элемента списка, так и нескольких. Список уровней английского кэшируется. После 1-го запроса выводится из кэша.

**События с**

Представляет собой календарь с возможностью выбора одной даты.

**1 – Заголовок**.

**2 – Кнопка «Назад»**.

**3 – Кнопка «Вперед»**.

**4 – Календарь**.

**5 – Кнопка «Отменить»**.

**6 – Кнопка «Применить»**.

**События по**

То же самое, что и [События с](#_События_с).

**Грейд**

**1 – Заголовок**.

**2 – Кнопка «Отмена»**.

**3 – Кнопка «Готово»**

**4 – Список всех грейдов**.

Список грейдов кэшируется. После 1-го запроса выводится из кэша.

**Последний рекрутер**

API: [get/employee/shortStaff](#_get/employee/shortStaff_1).

**1 – Заголовок**.

**2 – Кнопка «Отмена»**

**3 – Кнопка «Готово»**

**4 – Поисковая строка**: позволяет пользователю искать рекрутеров по ключевым словам. Поиск аналогичен п. [Желаемая позиция](#_Желаемая_позиция)

**5 – Список всех рекрутеров (всех сотрудников)**.

Возможен выбор как одного элемента списка, так и нескольких.

**HR доска. Главная**

**Термины и определения**

* HR доска **–** это раздел, содержащий список соискателей с назначенной им вакансий и соответствующим статусом.

В контексте данного ТЗ:

* Краткая информация соискателя – это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При нажатии на данный элемент происходит переход в карточку соискателя.
* Карточка соискателя – это элемент пользовательского интерфейса, представляющий собой блок подробной информации о соискателе в рамках раздела «HR доска», имеющий поле выбора ответственного HR, текстовое поле для комментария и кликабельный элемент, позволяющий перейти в профиль соискателя.
* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.

**API:**

1. **get/candidates/vacancies**

Метод предназначен для вывода краткой информации соискателя.

**Элементы интерфейса и логика работы**

**Экран загрузки**

1 – Кнопка «Выйти из ЛК».

2 – Название раздела.

3 – Кнопка «Применить фильтры»: позволяет рекрутеру отфильтровать список соискателей по необходимым параметрам.

Логика работы: при нажатии на данную кнопку открывается экран «Фильтры».

4 – Поисковая строка (логика описана ниже).

5 – Анимация загрузки.

6 – Нижняя навигационная панель.

**Главный экран**

**1 – Информационная строка «n соискателей»**: отображает количество соискателей, соответствующих:

* общему количеству соискателей при первоначальной загрузке экрана;
* количеству соискателей при применении фильтров;
* количеству соискателей при их сортировке посредством поисковой строки. В данном случае текст меняется на: «n соискателей найдено».

Логика работы: при первоначальной загрузке экрана отображается общее количество соискателей (например, «8062 соискателей»).

При сортировке соискателей при помощи строки поиска и/или применении фильтров появляется экран загрузки, затем текст обновляется, отображая количество соискателей, соответствующих текущим условиям.

Если соискатели не найдены, отображается текст «Ничего не найдено»

**2 – Краткая информация соискателя**: отображается список всех соискателей с краткой информацией.

Запрос: [get/candidates/vacancies](#_get/candidates/vacancies).

Применяется пагинация по 50 соискателей.

В краткой информации соискателя отображаются:

* Вакансия;
* Дата создания;
* Статус;
* ФИО соискателя;
* Город соискателя;
* Направление;

Аватар соискателя: для каждого соискателя загружается по id:

Метод GET;

URL: /userAvatar?id=

**3 – Кнопка «Удалить карточку соискателя»**. При нажатии на кнопку открывается экран удаления карточки соискателя

При возникновении ошибки при загрузке данных появляется сообщение. При нажатии на кнопку «Обновить» появляется экран загрузки, происходит повторный запрос [get/candidates/vacancies](#_get/candidates/vacancies).

**Реализация поиска:**

Строка поиска позволяет рекрутеру искать соискателей по ФИО.

Строка поиска представляет собой текстовое поле: разрешены буквы (кириллица и латиница), цифры, пробелы и специальные символы.

Логика работы: поиск выполняется по ключевым словам, содержащимся в ФИО.

Если пользователь ввел текст в строку поиска, а затем полностью стер его, система должна обработать этот случай как пустой запрос. В этом случае отображаются все соискатели без фильтрации.

Рекрутер вводит текст в строку поиска. Поиск не чувствителен к регистру. После ввода необходимого запроса рекрутер нажимает на кнопку «Готово». Выводится список соискателей

**HR доска. Фильтры**

**Термины и определения**

* HR доска **–** это раздел, содержащий список соискателей с назначенной им вакансий и соответствующим статусом.

В контексте данного ТЗ:

* Краткая информация соискателя – это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При нажатии на данный элемент происходит переход в карточку соискателя.
* Карточка соискателя – это элемент пользовательского интерфейса, представляющий собой блок подробной информации о соискателе в рамках раздела «HR доска», имеющий поле выбора ответственного HR, текстовое поле для комментария и кликабельный элемент, позволяющий перейти в профиль соискателя.
* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.

**API:**

1. **get/candidates/vacancies**

Метод предназначен для вывода краткой информации соискателя.

1. **get /hiring/vacancies**

Метод предназначен для вывода списка вакансий.

1. **get/dictionary/****candidateStatuses**

Метод по получению статусов соискателей для HR доски.

1. **get/hiring/departments**

Метод получения подразделений.

1. **get/dictionary/city**

Метод получения списка городов.

**Элементы интерфейса и логика работы**

**1 – Top bar:**

**2 – Меню фильтров.** Состоит из разных типов элементов:

**c – Строка навигации.** При нажатии на данную строку происходит переход на экран выбора параметра / параметров из списка.

**d – Переключатель.** Выбор между двумя значениями: «Да» / «Нет».

**3 – Кнопка «Применить»**: зафиксирована в нижней части экрана. Если ни один параметр не настроен, то кнопка неактивна.

**4 – Кнопка «Сбросить все фильтры»**: при нажатии на данную кнопку появляется диалоговое окно

При нажатии на кнопку «Сбросить» все фильтры сбрасываются

**5 – Выбранные параметры фильтрации:** при нажатии крестик выбранный параметр удаляется.

Можно настраивать множество параметров, т.е. можно выбрать несколько вакансий, статусов, подразделений, городов.

**Вакансия**

API: [get /hiring/vacancies](#_get_/hiring/vacancies).

1 – Заголовок:

2 – Кнопка «Отмена»

3 – Кнопка «Готово»

4 – Поисковая строка: позволяет рекрутеру искать вакансию по ключевым словам.

Логика работы: поиск выполняется по ключевым словам, содержащимся в названии вакансии. Рекрутер вводит текст в строку поиска. Поиск не чувствителен к регистру.

**5 – Список всех вакансий.** Вакансии расположены в алфавитном порядке

Возможен выбор как одного элемента списка, так и нескольких

**Статус**

API: [get/dictionary/candidateStatuses](#_get/dictionary/candidateStatuses).

1 – Заголовок.

2 – Кнопка «Отмена»

3 – Кнопка «Готово»

4 – Список всех статусов.

Список статусов кэшируется. После 1-го запроса выводится из кэша.

**Подразделение**

API: [get/hiring/departments](#_get/hiring/departments_1).

1 – Заголовок.

2 – Кнопка «Отмена».

3 – Кнопка «Готово»

4 – Поисковая строка: позволяет рекрутеру искать подразделение по ключевым словам. Поиск аналогичен п. [Вакансия](#_Вакансия).

5 – Список всех подразделений.

Возможен выбор как одного элемента списка, так и нескольких

**Город**

API: [get/dictionary/city](#_get/dictionary/city_2).

1 – Заголовок:

2 – Кнопка «Отмена»:

3 – Кнопка «Готово»:

4 – Поисковая строка: позволяет рекрутеру искать город по ключевым словам. Поиск аналогичен п. [Вакансия](#_Вакансия).

5 – Список всех городов. Возможен выбор как одного элемента списка, так и нескольких.

Первоначально список выводится методом API и кэшируется. При дальнейших запросах список выводится из кэша.

**HR доска. Карточка соискателя**

**Термины и определения**

* HR доска **–** это раздел, содержащий список соискателей с назначенной им вакансий и соответствующим статусом.

В контексте данного ТЗ:

* Краткая информация соискателя – это элемент пользовательского интерфейса, представляющий собой кликабельный блок с краткой информацией о соискателе. Включает ключевые данные, такие как ФИО, город, желаемую позицию, статус, телефон и аватар соискателя. При нажатии на данный элемент происходит переход в карточку соискателя.
* Карточка соискателя – это элемент пользовательского интерфейса, представляющий собой блок подробной информации о соискателе в рамках раздела «HR доска», имеющий поле выбора ответственного HR, текстовое поле для комментария и кликабельный элемент, позволяющий перейти в профиль соискателя.
* Рекрутер – это HR-специалист с назначенными доступами рекрутера и HR-менеджера.
* Сорсер – это HR-специалист с назначенным доступом рекрутера.

**API:**

1. **get/candidates/vacancies**

Метод предназначен для вывода краткой информации соискателя.

1. **get/employee/shortStaff**

Данный метод выводит краткую информацию обо всех работниках.

1. [**get**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Candidate_management/getCardComment)**/candidates/card/comment/{cardId}**

Метод предназначен для вывода комментария при открытии карточки соискателя.

1. **[put](http://smartstaff-dev.simbirsoft:8080/swagger/" \l "!/Candidate_management/updateCardComment) /candidates/card/comment/{cardId}**

Метод предназначен для обновления комментария в карточке соискателя.

1. **[put](http://smartstaff-dev.simbirsoft:8080/swagger/" \l "!/Candidate_management/updateCardHR)**[**/candidates/card/hr/{cardId}**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Candidate_management/updateCardHR)

Метод предназначен для обновления HR, за которым закреплен соискатель.

1. **get/dictionary/candidateStatuses**

Метод по получению статусов соискателей для HR доски.

1. [**put**](http://smartstaff-dev.simbirsoft:8080/swagger/#!/Candidate_management/updateCardStatus)**/candidates/status**

Метод предназначен для обновления статуса соискателя.

1. **put /candidates/card/{cardId}**

Метод предназначен для удаления карточки соискателя с HR доски.

1. **get/dictionary/status**

Метод получения списка статусов.

**Элементы интерфейса и логика работы**

**Карточка соискателя**

При нажатии на блок с краткой информацией соискателя на главном экране раздела «HR доска» открывается экран карточки соискателя.

**1 – Информация о соискателе.** Выводится по [get/candidates/vacancies](#_get/candidates/vacancies_1). Логика вывода данных:

* Вакансия
* Дата создания
* Статус
* ФИО соискателя
* Город соискателя
* Направление
* Аватар соискателя: для каждого соискателя загружается по id:
* Метод GET
* URL: /userAvatar?id=

**2 – Интерактивный элемент интерфейса.** При нажатии на данный элемент происходит переход в профиль соискателя.

**3 – Интерактивный элемент редактирования статуса.** При нажатии на строку «Статус» появляется экран выбора статуса соискателя.

**4 – Поле выбора HR-специалиста.** HR-специалист выводится сразу, либо отображается поле выбора. При нажатии на данное поле открывается экран выбора HR-специалиста.

**5 – Текстовое поле для комментария.** Сразу после запроса [get/candidates/vacancies](#_get/candidates/vacancies_1) делается дозапрос на комментарий [get /candidates/card/comment/{cardId}](#_get_/candidates/card/comment/{cardI).

**6 – Счетчик символов.** При вводе символов обновляется в реальном времени.

**7 – Кнопка «Сохранить».** Изначально неактивна. При выборе нового HR-специалиста и/или вводе нового комментария становится активной. При нажатии на данную кнопку обновленные данные карточки соискателя сохраняются [put /candidates/card/comment/{cardId}](#_put_/candidates/card/comment/{cardI), [put /candidates/card/hr/{cardId}](#_put_/candidates/card/hr/{cardId}), на главном экране раздела после экрана загрузки появляется сообщение

**8 – Кнопка «Закрыть».** При нажатии на данную кнопку никакие изменения не сохраняются, происходит возврат на главный экран раздела «HR доска»

**9 – Кнопка «Удалить HR».** При нажатии на данную кнопку выбранный HR сбрасывается, появляется поле выбора.

**Экран выбора HR-специалиста**

1 – Заголовок.

2 – Кнопка «Отмена».

3 – Кнопка «Готово»

4 – Поисковая строка: позволяет пользователю искать HR-специалиста по ключевым словам.

Логика работы: поиск выполняется по ключевым словам, содержащимся в ФИО сотрудника. Рекрутер вводит текст в строку поиска. Поиск не чувствителен к регистру.

**5 – Список всех сотрудников.**

API: [get/employee/shortStaff](#_get/employee/shortStaff_2).

Имя и фамилия сотрудников выводятся в алфавитном порядке (А-Я). Сотрудники должны быть действующие.

Возможен выбор только одного сотрудника

**Экран выбора статуса соискателя**

1 – Заголовок:

2 – Кнопка «Отмена»: при нажатии на данную кнопку происходит возврат к экрану Рис.2, выбранный параметр не сохраняется.

3 – Кнопка «Готово»: при нажатии на данную кнопку выбранный статус сохраняется и отображается заместо прежнего статуса. API: [put /candidates/status](#_put_/candidates/status).

Если отмечено значение «Не выбрано», то отображается прежний статус без изменений, как на Рис.2.

4 – Список всех статусов.

API: [get/dictionary/candidateStatuses](#_get/dictionary/candidateStatuses_1).

Возможен выбор только одного статуса. По умолчанию отмечен статус, установленный в карточке соискателя. Статусы соискателя кэшируются после первого запроса.

**Удаление карточки соискателя**

1 – Заголовок.

2 – Интерактивный элемент интерфейса. При нажатии на данный элемент происходит переход в профиль соискателя.

3 – Информация о соискателе. Выводится по [get/candidates/vacancies](#_get/candidates/vacancies_1).

* Вакансия;
* Дата создания
* Статус.
* Закреплен за: «hr»
* ФИО соискателя;
* Город соискателя.
* Направление.
* Аватар соискателя: для каждого соискателя загружается по id:
* Метод GET
* URL: /userAvatar?id=.

4 – Поле выбора. При нажатии на поле происходит переход на экран выбора статуса.

5 – Кнопка «Удалить». Изначально неактивна. При выборе статуса становится активной. При успешном удалении карточки соискателя с HR доски на главном экране появляется сообщение

6 – Кнопка «Закрыть». При нажатии на данную кнопку никакие изменения не сохраняются, происходит возврат на главный экран

**Экран выбора статуса**

1 – Заголовок.

2 – Кнопка «Отмена».

3 – Кнопка «Готово»

4 – Список всех статусов.

API: [get/dictionary/status](#_get/dictionary/status_1).

Возможен выбор только одного статуса. По умолчанию отмечено «Не выбрано». Статусы соискателя кэшируются после первого запроса.

2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ

Согласно индустриальным стандартам, работа конкурсанта должна быть сохранена на удаленном сервере с применением системы контроля версий. Доступ к системе и аккаунт выдается организаторами чемпионата. При невозможности скомпилировать приложение из предоставленных файлов, результат работы конкурсанта не может быть оценен. В случае разбития задания на сессии, для проверки используется версия, сохраненная конкурсантом в системе контроля версий только во время соответствующей сессии.

Задание может выполняться с использованием предоставленным и согласованным с Главным экспертом в день ознакомления с рабочим местом списком библиотек/плагинов, которые необходимо загружать через Интернет.

2.1. Личный инструмент конкурсанта

* Допускается использование личного инструмента для разработки по согласованию с ГЭ и предварительной проверки оборудования группой экспертов не аффилированных с участником. Инструмент при этом оставляется на площадке до момента завершения чемпионата.
* Конкурсанты могут использовать защиту для ушей
* Конкурсанты могут принести с собой свои клавиатуры, мышки и коврики для мышек. Все принесенные клавиатуры, мышки и коврики должны быть предварительно сданы на проверку технической команде. Устройства ввода не должны быть программируемыми.
* В случае невозможности полноценно обеспечить площадку мобильными устройствами, конкурсанты должны принести на площадку мобильные устройства согласно инфраструктурному листу. Данные мобильные устройства будут сброшены до состояния заводских настроек перед чемпионатом. Данные устройства будут находиться на территории площадки до окончания чемпионата.
  1. Материалы, оборудование и инструменты,

запрещенные на площадке

Конкурсанты могут слушать музыку. Наушники и музыка в виде файлов должна быть предварительно сдана в техническую команду для проверки. Принесенная музыка будет хранится на серверах для конкурсантов к которым они будут иметь доступ.

Конкурсант не имеет права приносить:

* + дополнительное программное обеспечение;
  + любые портативные средства связи, например, мобильные телефоны или интеллектуальные часы;
  + портативные цифровые приборы (планшеты, электронные помощники и т. д.);
  + внешние запоминающие устройства (карты памяти, флэшки и т. д.).

Оборудование не должно иметь доступ к встроенным устройствам хранения данных. Организатор соревнования обеспечит, чтобы они были отключены.

Эксперты имеют право запретить использование оборудования, принесенного на соревнование.

Конкурсантам может быть разрешен доступ в Интернет в зоне проведения соревнования. Для этого будет использоваться выделенный компьютер, доступ **будет ограничен 10-минутами на конкурсанта за одну сессию в порядке живой очереди, не чаще одного раза в 10 минут (Фиксация времени осуществляется поминутно).**

Экспертам разрешается пользоваться личными компьютерами, планшетами или мобильными телефонами, находясь в помещении для экспертов, за исключением случаев, когда документы, относящиеся к  соревнованию, находятся в комнате.

Экспертам разрешается пользоваться фото- и видеооборудованием, находясь в помещении для экспертов, за исключением случаев, когда документы, относящиеся к соревнованию, находятся в комнате, по согласованию с Главным экспертом.

Конкурсантам и экспертам разрешается использовать личные устройства для фото- и видеосъемки на рабочей площадке только после завершения конкурса.

3. Приложения

Приложение 1. Инструкция по заполнению матрицы конкурсного задания.

Приложение 2. Матрица конкурсного задания.

Приложение 3. Инструкция по охране труда.