

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Волгоградский государственный технический университет»

На правах рукописи



Гурьев Владислав Витальевич

АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ
ДЛЯ ЛЮДЕЙ С ОГРАНИЧЕННЫМИ ИНТЕЛЛЕКТУАЛЬНЫМИ
ВОЗМОЖНОСТЯМИ

2.3.7 – Компьютерное моделирование и автоматизация проектирования

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель
доктор технических наук,
профессор Садовникова Наталья Петровна

Волгоград, 2025

Оглавление

Введение.....	4
1. Анализ способов разработки МП	10
1.1 Подходы к разработке МП.....	10
1.2 Модель ЖЦ МП.....	11
1.3 Программные средства для поддержки ЖЦ МП	13
1.4 Особенности разработки МП для пользователей с ИН	19
1.4.1 Характеристики категории пользователей с ИН	19
1.4.2 Проблемы обеспечения доступности МП для пользователей с ИН.....	22
1.4.3 Способы разработки интерфейсов МП для пользователей с ИН	26
1.5 Выводы.....	28
2. Разработка метода автоматизированного проектирования МП для пользователей с ИН.....	30
2.1 Модель ЖЦ МП с адаптируемым интерфейсом.....	30
2.2 Концепция сквозной поддержки этапов ЖЦ МП.....	32
2.3 Метод проектирования интерфейса МП.....	33
2.4 Шаблон проектирования МП.....	36
2.5 Выводы.....	42
3 Разработка CASE-системы поддержки ЖЦ МП для пользователей с ИН.....	44
3.1 Архитектура системы	44
3.2 Функционал системы.....	46
3.3 Конфигурационная панель для адаптации интерфейса МП.....	48
3.4 Способ реализации многомодульной архитектуры МП	53
3.5 Интерфейс системы	56
3.6 Методика разработки МП с применением системы.....	60
3.7 Онтология проектирования МП для пользователей с ИН.....	63
3.8 Выводы.....	64
4. Апробация системы	66
4.1 Апробация системы на разработке МП с адаптируемым интерфейсом	67
4.1.1 Разработка МП для тестирования предпочтений пользователей	67
4.1.2 Разработка МП для тренировки навыков совершения покупок	68

4.1.3 Разработка МП для тренировки навыков различения предметов.....	72
4.2 Оценка временных затрат на разработку МП в системе	80
4.3 Оценка доступности МП	85
4.4 Выводы.....	87
Заключение	88
Список литературы	91
Приложение А. Результаты тестирования МП	107
Приложение Б. Свидетельство о регистрации программы для ЭВМ	109
Приложение В. Акт внедрения	110

Введение

Актуальность темы исследования. Непрерывный рост сложности и разнообразия программного обеспечения (ПО) обуславливает постоянно растущий интерес к автоматизированным системам, предназначенным для поддержки различных этапов жизненного цикла (ЖЦ) ПО. Программные средства автоматизированной разработки ПО (Computer-Aided Software Engineering, CASE) используются для анализа, проектирования, программирования, тестирования и испытаний, управления работоспособностью ПО. Одним из актуальных направлений исследований в сфере автоматизации разработки ПО является создание специализированных CASE систем, предназначенных для разработки прикладного ПО определенного вида и назначения.

В настоящее время на рынке ПО наблюдается рост приложений для людей с интеллектуальными нарушениями (ИН). Большая часть таких приложений реализуется на мобильной платформе, так как мобильные приложения (МП) просты в освоении и предоставляют широкие возможности для использования без привязки к месту.

Разработкой МП для людей с ИН занимаются не только специализированные компании, но и специалисты, профессиональная деятельность которых связана с обеспечением качества жизни людей с ИН. Такие специалисты не всегда обладают достаточно высокой квалификацией разработчиков МП, а это сказывается как на скорости разработки таких приложений, так и на их качестве.

Ключевым критерием качества МП для пользователей с ИН является доступность. В соответствии с действующим в настоящее время в РФ ГОСТ (Р 52872-2019) все разрабатываемое ПО должно обеспечивать доступность для различных категорий пользователей с ограниченными возможностями, в том числе для людей с нарушениями ментальной сферы. В то же время к настоящему времени по данным Роскачества до 95% разрабатываемых МП не адаптированы для людей с ограниченными возможностями здоровья.

Особенности требований к МП для людей с ИН и специфика их разработки приводят к необходимости выделения таких МП в самостоятельную категорию ПО. Для разработки МП, доступных для пользователей с ИН, требуются специализированные CASE-средства, минимизирующие квалификационные требования к разработчику МП и трудоёмкость их разработки и повышающие при этом качество самих МП.

Степень разработанности темы. Различные подходы к моделированию ЖЦ ПО рассматриваются в работах М. Hossain, A. Alazzawi, T. Vithani, A. Anandkumar, Е.А. Искра, А. Kaur, К. Kaur, А.А. Вичугова. Разработке моделей и методов автоматизированного проектирования МП посвящены исследования М. Yamin, К. Basel, G. Kapila, A. Dias, L.F. da Silva, P.A. Parreira, A.P. Freire, Е.А.Искра, R. Bennett, R. Vijaygopal, P. Korczak, A. Zwierzchowska, A. Alhanouf, A. Khalid., C. Juan, W. Ida, J. Letizia, P. Susanna, C. Stefano, G. Javier, H. Gunnar, M. Henriette. Исследованиями проблем разработки ПО для людей с ИН занимаются J. Dekelver, M. Braun, M. Wölfel, G. Renner, C. Menschik, P. Heumader, K. Miesenberger, T. Murillo-Morales, D. Miraz, M. Ali, P. Excell, О.А. Корнева. Способам обеспечения доступности МП посвящены работы Ballantyne, M. Jha, A. Jacobsen, A. Hawker, J.S. El-Glaly, Y.N. Acosta-Vargas, P. Salvador-Ullauri, L. Jadán-Guerrero, J. Guevara, C. Sanchez-Gordon, S. Calle-Jimenez, T. Lara-Alvarez, P. Medina, A. Nunes, I.L Choi, H. Park, J.W. Teixeira, P. Eusebio, C. Teixeira, Leonor. Модели, методы и технологии обеспечения доступности МП представлены в работах О. А. Шабалиной, С. Perez, М. Б. Кульцовой, А. С. Поцелуйко, И. Г. Жуковой, Д. С. Матюшечкина.

Объектом исследования является разработка МП для пользователей с ИН.

Предметом исследования являются способы автоматизации проектирования МП для пользователей с ИН.

Методы исследования. В диссертационном исследовании использованы методы автоматизации проектирования, методы моделирования GUI, методы человеко-компьютерного взаимодействия (Human-computer interaction), методы

на основе шаблона (Pattern-Based Design), методы итерационного дизайна (Iterative Design), методы ориентированного на пользователя проектирования (User-Centered Design), A/B методы анализа пользовательского опыта и тестирования.

Целью диссертационной работы является повышение качества МП для пользователей с ИН и снижение трудоемкости разработки МП за счет автоматизации проектных и технологических процедур создания адаптируемых интерфейсов и архитектуры МП. Для достижения поставленной цели выделены следующие задачи:

- 1) провести анализ моделей, методов, технологий автоматизированной разработки МП и возможностей применения существующих решений к разработке МП для пользователей с ИН;
- 2) разработать метод автоматизированного проектирования МП для пользователей с ИН, обеспечивающий сквозную поддержку этапов ЖЦ МП;
- 3) спроектировать и реализовать CASE-систему для поддержки ЖЦ МП;
- 4) провести тестирование CASE-системы и оценить трудоемкость разработки и доступность МП для пользователей с ИН.

Научная новизна заключается в разработке метода автоматизированного проектирования МП для пользователей с ИН, в рамках которого получены следующие новые научные результаты:

- 1) предложена модель ЖЦ МП для пользователей с ИН как нового класса ПО, отличающаяся составом проектных и технологических процедур, обеспечивающих поддержку сквозной разработки этапов «Проектирование – Производство – Эксплуатация» МП для пользователей с ИН (п.10);
- 2) разработан шаблон проектирования МП на основе многомодульной архитектуры, отличающийся слойным делением модулей, что позволяет сократить количество модулей в архитектуре МП и снизить трудоемкость проектирования МП для пользователей с ИН (п.3);

- 3) впервые предложен метод автоматизированного проектирования интерфейсов МП для пользователей с ИН на основе адаптируемых шаблонов, автоматически генерируемых с помощью программно-реализуемой конфигурационной панели, обеспечивающий возможность структурной и параметрической адаптации интерфейсов МП под конечного пользователя с ИН на этапе эксплуатации без изменения программного кода МП (п.3).

Теоретическая и практическая значимость.

Теоретическая значимость проведенных исследований заключается в развитии методов автоматизированного проектирования специализированного ПО. Предложенный подход к автоматизации проектирования МП для пользователей с ИН может быть использован как методологический базис для создания новых моделей и методов проектирования ПО, примененного в разных прикладных областях и для разных категорий пользователей.

Практическая значимость работы заключается в создании CASE-системы, обеспечивающей поддержку разработки МП с адаптируемым интерфейсом, доступных для пользователей с различными возможностями и ограничениями. Апробация системы на разработке МП с адаптируемым интерфейсом показала возможность применения системы в компаниях, занимающихся разработкой адаптируемого МП. Тестирование МП, разработанных в системе и адаптированных под конечных пользователей в Волгоградском психоневрологическом интернате и медицинском центре MPI Oosterlo (Бельгия) подтвердила возможность использования системы специалистами, профессиональная деятельность которых связана с поддержкой людей с ИН. Система апробирована на кафедре САПР и ПК ВолгГТУ в рамках проведения летней практики студентов 2, 3 курса обучения и реализации выпускных квалификационных работ бакалавров.

Апробация результатов исследования проводилась в период стажировки в университете Томаса Мора, г. Гиль, Бельгия в рамках Международного проекта

“Assistive technologies for people with disabilities” по программе Erasmus+ “Mobility projects for higher education students and staff”.

Основные положения, выносимые на защиту:

- 1) Модель ЖЦ МП с расширенным составом этапов и процедур, применимая для разработки МП для пользователей с ИН.
- 2) Шаблон проектирования МП с адаптируемым интерфейсом на основе многомодульной шаблонной архитектуры со слойным делением модулей и повторным использованием типовых модулей, сокращающий трудоемкость разработки МП.
- 3) Метод проектирования интерфейса МП на основе автоматически генерируемых адаптируемых шаблонов, обеспечивающая возможность структурной и параметрической адаптации интерфейса МП без изменения программного кода.
- 4) CASE-система поддержки ЖЦ МП, реализующая интеграцию методов и технологий разработки МП с адаптируемым интерфейсом.

Апробация. По материалам диссертации опубликовано 11 статей, из них 3 статьи в рецензируемых научных изданиях, рекомендованных ВАК, 4 статьи, зарегистрированные в базах Scopus и Web of Science, 4 публикации в сборниках трудов конференций. Получено 2 свидетельства о регистрации программы для ЭВМ.

Основные положения и материалы диссертационной работы докладывались на III Международной научно-практической конференции “Creativity in Intelligent Technologies and Data Science Third Conference, CIT&DS” в 2019 году, XI Международной конференции “International Conference on Information, Intelligence, Systems and Applications” в 2020 году, V Международной научно-практической конференции “Creativity in Intelligent Technologies and Data Science Conference, CIT&DS” в 2023 году, VIII Всероссийской научной конференции с международным участием "Информационные технологии в моделировании и управлении: подходы, методы, решения" в 2025 году, на XXXVIII Международной научной

конференция “Математические Методы в Технике и Технологиях”, ММТТ-38 в 2025 году, на V международной научной конференции “Информационные технологии в науке, управлении, социальной сфере и медицине” в 2025 году.

Личный вклад автора. В диссертации представлены результаты исследований, выполненных самим автором. Личный вклад автора состоит в постановке задач исследования, разработке теоретических и прикладных методов их решения, в обработке, анализе, обобщении полученных результатов и формулировке выводов.

Структура и объем диссертации. Диссертация изложена на 110 страницах, содержит 48 рисунков и 9 таблиц, включает введение, 4 главы, заключение, список литературы из 139 источников, 2 приложения.

Благодарности. Автор благодарит студентов и аспирантов кафедры «Системы автоматизированного проектирования и поискового конструирования» ВолгГТУ за участие в разработке МП для пользователей с ИН, а также сотрудников Специализированного медицинского центра MPI Oosterlo, Бельгия и Волгоградского психоневрологического интерната, Россия за организацию тестирования CASE-системы и МП, разработанных с применением системы, с клиентами с ИН и активное обсуждение результатов. Особую благодарность выражает Шабалиной Ольге Аркадьевне, кандидату технических наук, доценту кафедры «САПР и ПК» ВолгГТУ, за консультации в ходе выполнения диссертационной работы.

1. Анализ способов разработки МП

1.1 Подходы к разработке МП

В настоящее время основными подходами к разработке МП [15] являются: нативный, кроссплатформенный, гибридные веб-приложения, прогрессивные веб-приложения. Нативные приложения представляют собой МП, написанные на языке программирования и в средах, предоставляемых владельцем платформы и работающих непосредственно в операционной системе устройства, такой как iOS или Android. Нативные приложения обычно разрабатываются в интегрированной среде разработки (IDE). Такие среды разработки предоставляют разработчикам различные средства для настроек отладки, управления проектом, контроля версии и другие необходимые инструменты [16,17]. Нативные приложения позволяют создавать более сложные и визуально привлекательные пользовательские интерфейсы [18].

Кроссплатформенные приложения могут быть написаны на разных языках программирования и в разных средах, но они компилируются в приложение, что обеспечивает функционирование на различных аппаратных платформах и операционных системах [19]. Исследования показывают, что адаптация кроссплатформенных подходов улучшает продуктивность команды разработчиков и сокращает затраты на поддержку и обновление приложений [20-22].

Гибридные веб-приложения [23] разрабатываются с использованием веб-технологий, таких как JavaScript, CSS и HTML5, и объединенных в пакеты установки приложения. В отличие от нативных приложений, гибридные приложения работают в «веб-контейнере», который обеспечивает среду выполнения браузера и API-интерфейс для нативных устройств через Apache Cordova. С научной точки зрения, гибридные веб-приложения можно рассматривать как пример концепции "разделенного кода". Исследования показывают, что такие подходы не только повышают эффективность работы команд разработчиков, но и приводят к лучшему пользовательскому опыту, так

как пользователи получают доступ к приложениям на разных устройствах без значительных потерь в функциональности и производительности [24].

Прогрессивные веб-приложения (Progressive Web Application, PWA) подразумевают под собой веб-приложения, которые используют набор возможностей браузера – например, работу в автономном режиме, запуск фонового процесса и добавление ссылки на домашний экран устройства - для обеспечения удобства работы приложения [25]. Исследования показывают [26], что PWA могут значительно улучшить пользовательский опыт, увеличивая вовлеченность и время, проведенное в приложении. Они также предлагают экономически эффективное решение для бизнеса, так как разработка PWA часто обходится дешевле, чем создание отдельных нативных приложений для каждой платформы [27-29].

1.2 Модель ЖЦ МП

В условиях непрерывного роста сложности и разнообразия ПО эффективность процесса разработки ПО и качество самого ПО в большой степени зависит от выбора модели ЖЦ ПО [30, 31]. Под ЖЦ ПО понимают период времени, который начинается с момента принятия решения о создании программного продукта и заканчивается в момент его полного изъятия из эксплуатации. В ЖЦ ПО принято выделять следующие этапы [32,33]: анализ требований, проектирование, реализация, тестирование и отладка, эксплуатация и сопровождение.

Порядок и состав процедур этапов ЖЦ ПО принято описывать в виде модели ЖЦ ПО (Software development Life Cycle, SDLC), содержащей описание процессов, действий и задач, которые осуществляются в ходе разработки, использования и сопровождения программного продукта [34,35]. Особое внимание в процессе разработки ПО уделяется начальным этапам – анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов.

Производительность команды разработчиков и качество ПО зависят от эффективности определения и анализа показателей процесса разработки программного обеспечения на протяжении всего ЖЦ.

Выбор программных средств для поддержки каждого этапа ЖЦ может варьироваться в зависимости от требований от заказчика к ПО, возможностей и задач команды разработки. Так, в [36] рассмотрены метрики этапа проектирования, которые могут варьироваться в зависимости от методологии и цели компании. Анализ моделей ЖЦ ПО, их возможности и ограничений представлен в [37]. Выбор правильной модели ЖЦ имеет решающее значение в продуктовой отрасли, поскольку продукт должен быть доставлен до клиента в установленный срок и также должен обладать необходимым качеством. В [30] рассмотрены критерии для моделей разработки ПО и приведен математически обоснованный выбор рациональной модели ЖЦ разработки ПО в зависимости от важности критериев.

Исследования возможности применения моделей ЖЦ ПО для разработки МП показывают [38-39], что классические и традиционные SDLC модели и их методы не могут быть применены непосредственно к разработке МП без существенных изменений. Обоснованием таких заключений являются характеристики, специфичные для разработки МП. Поэтому в современных исследованиях модель ЖЦ мобильных приложений (Mobile Application Development Lifecycle, MADLC) рассматривается как самостоятельная категория моделей ЖЦ ПО, отличающаяся содержанием и даже количеством этапов [40,41]. В [42] отмечено, что большое количество неудачных МП связано с тем, что разработчики МП не применяют концепцию ЖЦ для разработки МП. В [43] рассмотрены факторы, которые приводят к сокращению количества этапов ЖЦ МП при разработке МП и показаны последствия таких проектных решений.

Особенности этапа эксплуатации ЖЦ МП, обуславливающие необходимость модификации модели ЖЦ ПО, приведены в [44,45]:

- многозадачность;

- используемое оборудование и ресурсы (имеются в виду не только программные, но и аппаратные, в Android-системах нужно учесть всё многообразие разрешений экрана и форм факторов);

- имеющиеся ограничения и требования операционной системы (например, работа в фоне, список разрешений, запрашиваемых у пользователя, запуск своих служб и сервисов, доступ к специальным хранилищам данных, особенности API и т.п.), в том числе касающиеся вопросов безопасности.

Типовой состав процедур этапов ЖЦ МП [40] показан на Рисунке 1.1.

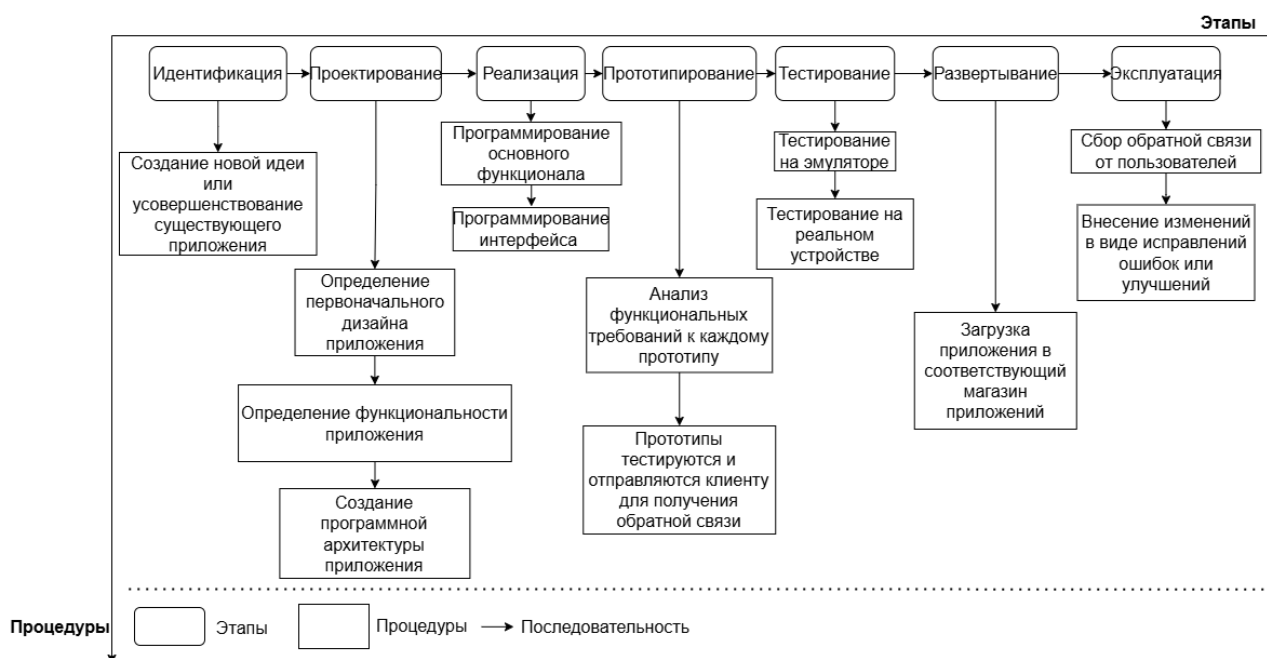


Рисунок 1.1 — Типовая MADLC

1.3 Программные средства для поддержки ЖЦ МП

В соответствии с действующим в настоящее время стандартом ISO/IEC 14102 [128] CASE-средства определяют как программные средства для поддержки процессов ЖЦ ПО. CASE-средства, доступные на рынке ПО, применяются на различных этапах ЖЦ для анализа, проектирования, программирования, тестирования и испытаний, а также для контроля работоспособности ПО [46-49].

Автоматизация разработки ПО предполагает наличие адекватных моделей ЖЦ ПО, включая технологии разработки и реализации, модели

архитектуры систем и их программных компонент [50,51]. Исследования показывают [52], что для каждого этапа ЖЦ применяют свои программные средства, направленные на выполнение конкретного круга задач. Применяемые программные средства на разных этапах ЖЦ ПО представлены в таблице 1.1.

Таблица 1.1 — Программные средства для поддержки ЖЦ ПО

Этап ЖЦ ПО	Программные средства	Примеры
Анализ требований (Requirements analysis)	Браузеры текстовые редакторы	Google Chrome Opera Microsoft Word
Проектирование (Application design)	САПР	AutoCAD PTC Creo
Реализация (Application development)	IDE	Android Studio PhpStorm PyCharm
Тестирование (Testing)	Графический пользовательский интерфейс, код программы	Браузер Библиотеки для юнит-тестирования
Эксплуатация (Using the app)	Браузер (для сбора статистики)	Яндекс.Метрика

Одним из актуальных направлений исследований в сфере автоматизации разработки ПО является создание специализированных CASE-средств, предназначенных для разработки ПО определенного назначения, таких как экспертные системы [53], PCS системы (Product Configuration System) [54],

системы бухгалтерского учета [55] и др. В последних исследованиях обсуждаются возможности поддержки ранних этапов ЖЦ ПО. CASE-системы, предназначенные для поддержки ранних этапов ЖЦ ПО определяют как самостоятельную категорию систем интеллектуальной компьютерной разработки ПО (I-CASE) [56]. Характеристики, специфические для МП, обуславливают необходимость разработки специализированных CASE-средств, обеспечивающих поддержку этапов ЖЦ МП, как самостоятельной категории ПО.

Для управления ЖЦ мобильных устройств применяется технология Mobile Device Management (MDM) [57]. Основной функционал MDM систем: учет используемых устройств; управление конфигурациями ОС; управление мобильными приложениями (в том числе их инициализация и деинициализация, удаленная очистка, удаленный мониторинг и контроль за сбойными ситуациями) [58-59].

Под средствами управления мобильными приложениями (Mobile Application Management, MAM) понимают ПО или услугу, отвечающие за управление и контроль доступа к разработанным внутри компании или коммерческим мобильным приложениям, используемые как в организации, так и на личных мобильных устройствах [60,61]. Управление мобильными приложениями позволяет администраторам применять в отношении мобильных устройств сотрудников внутренние политики организации. При этом термин MDM иногда используется для систем, в которых также реализованы функции управления МП.

В последнее время появился объединяющий термин EMM (Enterprise Mobility Management). [62], который определяет ПО, включающее MDM и MAM системы, и наборы серверного промежуточного ПО, позволяющего решать задачи интеграции мобильных приложений с корпоративными сервисами, например, задачу синхронизации файлов (Enterprise File Sharing) [63].

МAM-приложения отличаются от средств управления мобильными устройствами (MDM) тем, что последние фокусируются на управлении всем устройством и требуют, чтобы пользователи зарегистрировали свое устройство и установили сервисный агент. Хотя некоторые комплекты корпоративного управления мобильностью (EMM) включают функцию МAM, их возможности могут быть ограничены по сравнению с автономными решениями МAM, поскольку для комплектов EMM требуется профиль управления устройством для включения возможностей управления приложениями. Комплексное решение МAM предоставляет возможность: управлять установкой, обновлением и удалением мобильных приложений через корпоративный магазин приложений, отслеживать производительность и использование приложений и удаленно стирать данные из управляемых приложений.

Платформы разработки мобильных приложений MADP (Mobile Application Development Platform), относятся к комплексной группе услуг и продуктов, которые позволяют разрабатывать МП [64]. Типовой функционал MADP включает функции управления МП как при их размещении на сервисе для скачивания пользователями, так и в течение всего ЖЦ МП. На практике использование MADP означает, что компания может один раз разработать МП, а затем развернуть приложение на различных мобильных устройствах, включая планшеты, смартфоны и ноутбуки. MADP гарантирует, что МП совместимо с каждым устройством. MADP состоит из мобильного клиентского приложения и мобильного сервера промежуточного ПО. Сервер промежуточного ПО не хранит данные, но управляет данными посредством безопасности, системной интеграции, масштабируемости, связи, межплатформенной поддержки. Клиентские приложения подключаются к серверу и являются движущей силой бизнес-логики и пользовательского интерфейса на любом мобильном устройстве. Примерами таких систем могут служить мобильная версия платформы SAP Mobile Platform, IBM Mobile First [65], JAMF [66]. Платформа JAMF предназначена для комплексного управления мобильными устройствами Apple на предприятии. Система Okta [67] предоставляет IT-менеджерам

инструменты для управления всеми веб-приложениями, входящими в рабочую среду, с управлением по требованию и управлением доступом, работающими в облаке и за брандмауэром. Система Mobilock Pro [68] реализована как облачная платформа, позволяющая пользователям контролировать все устройства Android своих компаний Система ManageEngine Mobile Device Manager [69] позволяет управлять мобильными активами организации при сохранении надежной безопасности. Система Appery.io предназначена для разработки приложений с возможностью встраивания собственного кода: в системе имеется возможность создавать гибридные мобильные приложения, веб-приложения и PWA [70]. Система TheAppBuilder [71] представляет собой настольный сервис для PC, позволяющий создавать бесплатные приложения для платформ iOS и Android с возможностью добавления дополнительных функций после оплаты, не требует знаний языков программирования и использует визуальное программирование. Система iBuildApp [72], предоставляет возможность разработки МП с использованием визуального программирования, а также инструменты для управления МП. Система AppsBuilder [73] предоставляет возможность создания web-приложений с помощью визуального программирования.

Таким образом, для поддержки ЖЦ МП применяют различные CASE средства, отличающиеся назначением и функционалом [74]. На этапе реализации МП применяют MADM системы, для администрирования МП, разработанных для использования в различных видах профессиональной деятельности, и управления пользователями таких МП на этапе эксплуатации применяют MAM системы. Разработку мобильных приложений можно вести с использованием платформ MADP, которые в дополнение к функционалу MADM систем предоставляют некоторые функции MAM систем. Категории CASE средств поддержки ЖЦ МП приведены в таблице 1.2.

Таблица 1.2 — Категории CASE средств поддержки ЖЦ МП

Средство поддержки	Аббревиатура	Назначение	Этапы ЖЦ МП
Управление разработкой мобильных приложений	Mobile Application Development Management (MADM)	Администрирование МП, разработанных для использования в различных видах профессиональной деятельности	Реализация, развертывание, эксплуатация
Платформа разработки мобильных приложений	Mobile Application Development Platform (MADP)	Разработка МП	Реализация, развертывание, эксплуатация
Управление мобильными приложениями	Mobile Application Management (MAM)	Контроль доступа к МП	Эксплуатация

Большая часть средств поддержки ЖЦ МП, применяемых при разработке МП, поддерживают только один этап ЖЦ, выполнение остальных этапов ЖЦ реализуется в других системах или программах, не связанных между собой. Такие средства поддержки, как appery.io, TheAppBuilder, iBuildApp, поддерживают два этапа ЖЦ МП (реализация, эксплуатация). Однако такие системы ориентированы на разработку МП бизнес-направленности, соответственно, поддержка МП на этапе эксплуатации в основном заключается в сборе статистики, управлении правами доступа пользователей МП и других функций, предназначенных для администраторов МП.

Категории и примеры CASE средств для поддержки ЖЦ МП показаны на рисунке 1.2.

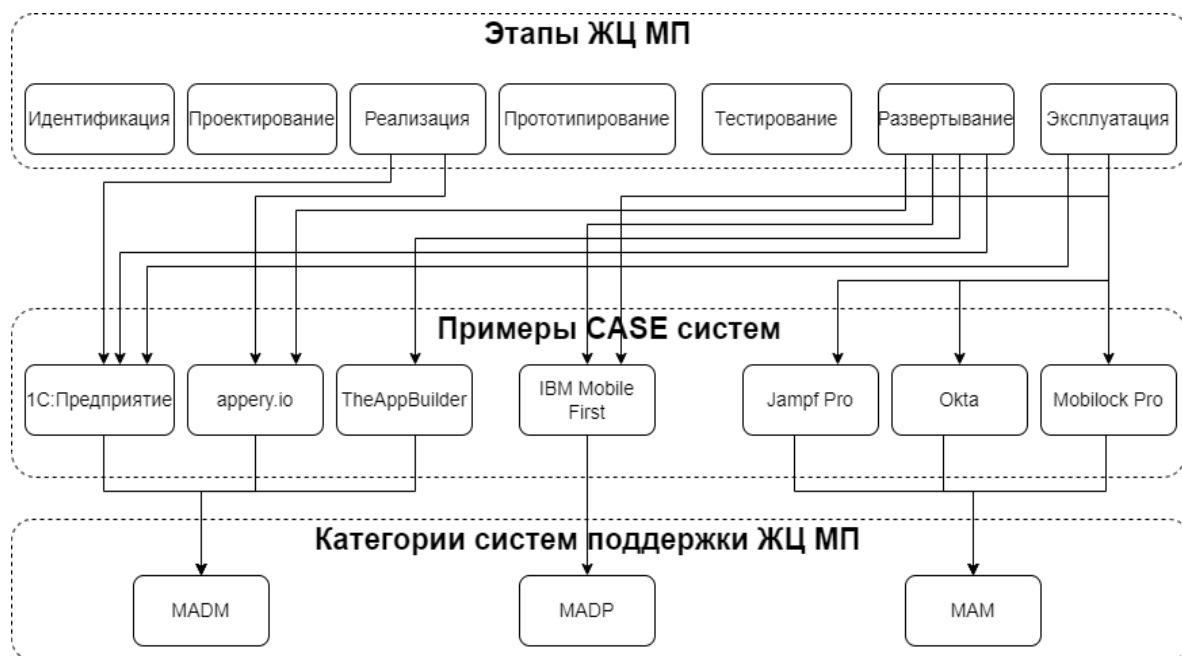


Рисунок 1.2 — Категории и примеры CASE средств для поддержки ЖЦ МП

1.4 Особенности разработки МП для пользователей с ИН

1.4.1 Характеристики категории пользователей с ИН

По данным Всемирной организации здравоохранения [1], почти 200 миллионов человек в мире имеют ИН, являясь самой большой группой среди населения с ограниченными возможностями здоровья. В России количество людей с ограниченными интеллектуальными возможностями приближается к миллиону [2]. Эпидемиологические исследования показывают, что общая распространенность тяжелых нарушений интеллекта (приблизительно $IQ < 50$) составляет от 0,3 до 0,4% людей всех возрастов, что означает, что в 27 странах Европейского Союза (общая численность населения 504 миллиона человек) от 1,5 до 2 миллионов человек имеют тяжелые нарушения интеллекта [76]. Метаанализ исследований, основанных на популяции, показал, что распространенность нарушений интеллекта во всех 52 исследованиях, включенных в метаанализ, составила 10,37/1000 населения [77].

Американская ассоциация по интеллектуальным и развивающимся нарушениям определяет интеллектуальную инвалидность как «инвалидность, характеризующуюся значительными ограничениями как в интеллектуальном

функционировании, так и в адаптивном поведении, которое охватывает ряд повседневных социальных и практических навыков. Эта инвалидность возникает до достижения 18-летнего возраста». [78]. Интеллектуальное функционирование относится к общим умственным способностям, таким как обучение, рассуждение, решение проблем и т. д. Адаптивное поведение — это совокупность концептуальных, социальных и практических навыков, которые изучаются и применяются людьми в повседневной жизни. Например: язык и грамотность, деньги, время и понятия чисел (концептуальные), межличностные навыки, социальная ответственность, самооценка, решение социальных проблем, способность следовать правилам/соблюдать законы и избегать преследования (социальные), уход за собой, профессиональные навыки, здравоохранение, путешествия/транспорт, расписания/рутины, безопасность, использование денег, использование телефона (практические) [79]. Определение Американской психиатрической ассоциации сопоставимо: «Интеллектуальная инвалидность (расстройство интеллектуального развития) — это расстройство, начинающееся в период развития, которое включает в себя как интеллектуальные, так и адаптивные функциональные дефициты в концептуальной, социальной и практической областях». [80]. Диагностический термин «Интеллектуальная инвалидность» (Intellectual Disability) является эквивалентным термином для диагноза «Расстройства интеллектуального развития» (Intellectual Developmental Disorders, IDD) по МКБ-11.

Международная классификация болезней (МКБ) является стандартным диагностическим инструментом для эпидемиологии, управления здравоохранением и клинических целей, как это определено Всемирной организацией здравоохранения [14]. МКБ используется врачами, медсестрами, другими поставщиками услуг, исследователями, менеджерами и кодировщиками медицинской информации, работниками медицинских информационных технологий, политиками, страховщиками и организациями пациентов для классификации заболеваний и других проблем со здоровьем, зафиксированных

во многих типах медицинских и жизненно важных записей, включая свидетельства о смерти и медицинские карты [75, 81].

Одним из ключевых обязательств современного общества является обеспечение полноценной качественной жизни людям с ИН, их социализация и обучение навыкам трудовой деятельности [3,82,83]. Перспективным направлением в этой сфере является внедрение в повседневную жизнь людей с ИН мобильных приложений (МП) [4,5], предназначенных для развития и тренировки различных навыков, необходимых в повседневной жизни, помогающих повысить самостоятельность и независимость людей с ИН [6-10].

Исследования показывают, что существующие ИТ существенно расширяют возможности для включения людей с ИН в социальную жизнь общества, и, в конечном итоге, повышения качества их жизни [84-88]. На рисунке 1.3 представлено использование смартфонов и действия людей с ограниченными возможностями [124].

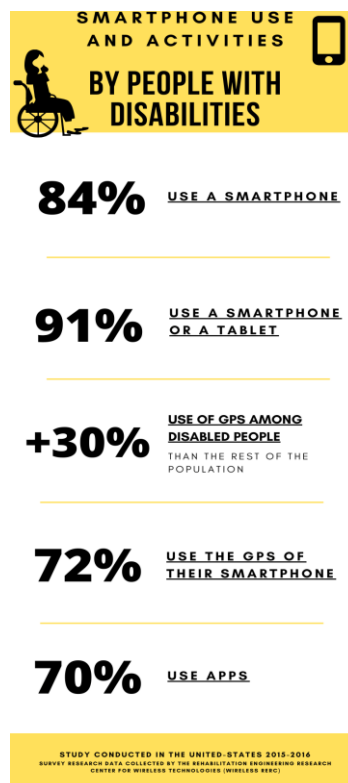


Рисунок 1.3 — Использование смартфонов и действия людей с ограниченными возможностями

1.4.2 Проблемы обеспечения доступности МП для пользователей с ИН

Направление исследований, посвященных анализу и решению проблем обеспечения доступности, активно развивается [85,89,90]. Тенденции роста интереса к проблеме обеспечения доступности МП показаны на рисунке 1.4 [125].

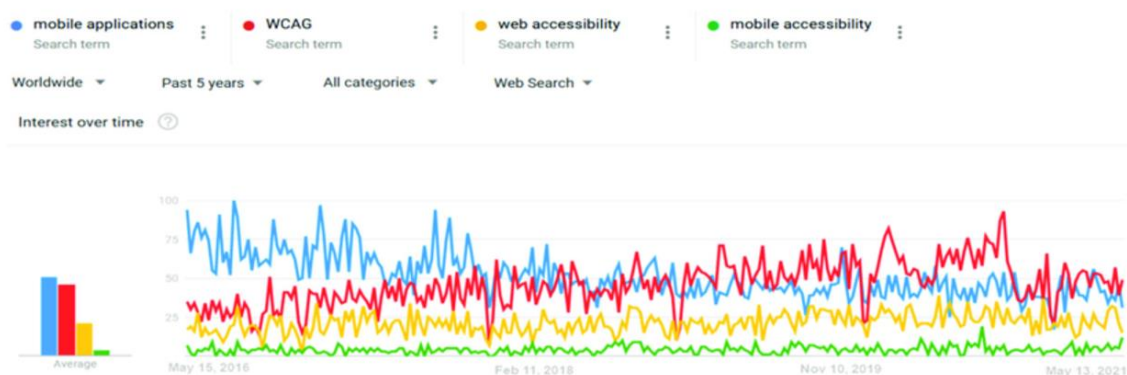


Рисунок 1.4 — Google Trends тенденции ключевых слов по теме “Обеспечение доступности МП” с 2016 по 2021 год

В [101] отмечено, что доступный дизайн фокусируется на предоставлении как можно большему числу людей доступа к программным продуктам и услугам. В работе изучено взаимодействие доступности с другими факторами, показано, что такие атрибуты качества как удобство использования и функциональность напрямую зависят от доступности ПО. В [102] отмечено, что возможность доступа к ПО в повседневной жизни жизненно важна для каждого, и поэтому доступность является фундаментальной проблемой для разработки ПО. Проведено исследование критических проблем и преимуществ включения доступности в разработку и проектирование ПО. Предложено рассматривать доступность как первостепенное соображение на протяжении всего процесса разработки ПО.

В [103] показано, что мобильные технологии могут помочь взрослым с ИН инициировать и поддерживать социальные связи в важных сферах их жизни, таких как семья, друзья и работа/волонтерство. В работе определено, как конкретные аспекты использования МП связаны с социальной включенностью

людей с ИН. Результаты исследования показали, что использование МП среди взрослых с ИН было положительно связано с их социальной включенностью в семью, друзей и работу/волонтерство. В [104] отмечено, что за последние несколько лет технологии, особенно мобильные, значительно продвинулись вперед что может предоставить прекрасную возможность для улучшения независимости людей ИН. Проведенное в работе исследование показало, использование МП позволяет поддержать процесс реабилитации человека с ИН.

В [105] исследованы возможности использования МП для развития социальных навыков, таких как коммуникативные навыки и поведенческие навыки для учащихся с ИН. Результаты этого исследования показывают, что в результате применения МП коммуникативные навыки и поведенческие навыки учащихся улучшились и имеют статистически значимое значение.

Особенности разработки МП для людей с ИН исследованы в [106]. Показано, что МП имеют большой потенциал для облегчения жизни людей с ИН. Фактически, можно разрабатывать специальные приложения, адаптированные к их потребностям. Но даже в этом случае пользователи могут испытывать проблемы с доступностью некоторых структурных элементов смартфонов.

Проблеме доступности посвящена работа [107]. В исследовании показано, что проблемы доступности в МП создают трудности для людей с ИН, что противоречит целям устойчивого развития и Конвенции о правах инвалидов. Целью статьи является исследование того, в какой степени отрасль учитывает доступность при разработке мобильных обучающих приложений для этой конкретной группы пользователей. Приведены результаты оценки доступности интерактивных МП, разработанных для чтения и изучения алфавита,

Анализ состояния дел в сфере разработки МП с учетом проблем доступности для людей с когнитивными, двигательными и сенсорными нарушениями, проведенный в [108] показал, что в несмотря на то, что количество разрабатываемых МП растет в геометрической прогрессии,

применение руководств по обеспечению доступности часто игнорируется разработчиками.

В [109] также отмечается, что МП должны быть доступны для всех пользователей. Основные результаты исследования показывают, что большинство рекомендаций по доступности игнорируются при разработке мобильных приложений. Такое поведение в основном обусловлено недостаточной осведомленностью разработчиков о проблемах доступности и отсутствием инструментов для поддержки разработки в процессе разработки.

В [110-112] рассмотрены методологии оценки доступности мобильных устройств и предложены рекомендации. Показано, что отсутствие руководств и стандартов, специально адаптированных для мобильных устройств, снижает эффективность оценки и мониторинга доступности.

В [114] предложена практическая структура для разработки адаптивных пользовательских интерфейсов, основанная на системе рекомендаций с учетом контекста, и интеграции различных компонентов для создания адаптивных пользовательских интерфейсов для цифровых услуг. В [115] показано, что стратегии адаптивного дизайна взаимодействия играют решающую роль в адаптации привычек поведения пользователя, способствуя улучшению пользовательского опыта и удовлетворенности. На рисунке 1.5 представлены компоненты цифровой доступности [113].

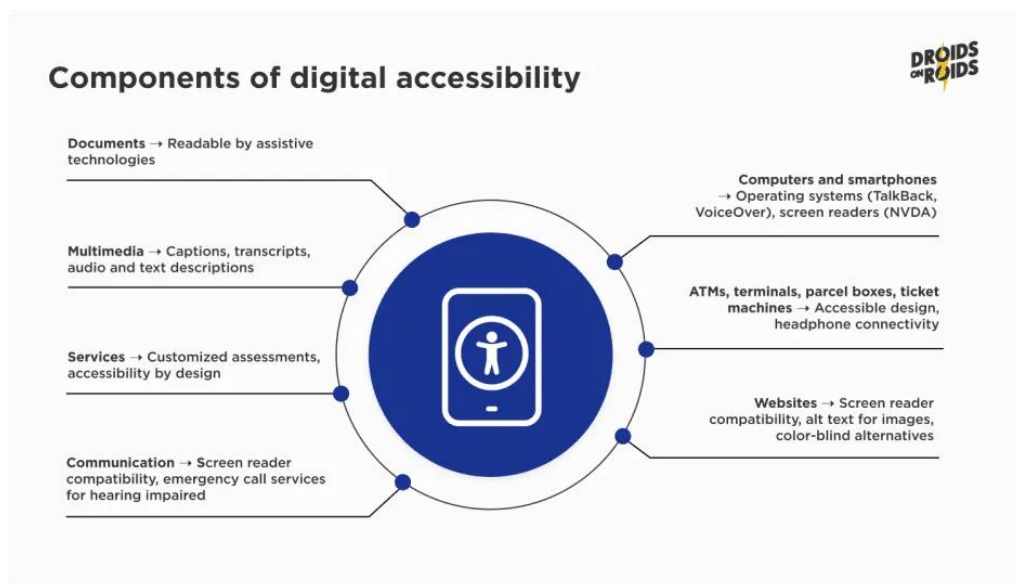


Рисунок 1.5 — Компоненты цифровой доступности

В Руководстве по доступности веб-контента (Web Content Accessibility Guidelines, W3C WAI) [126] приведены общие принципы обеспечения доступности, применимые к разработке МП:

- воспринимаемость (perceivability): любая информация, а также компоненты МП должны быть представлены максимально воспринимаемым способом;
- управляемость (Operability): любые компоненты интерфейса приложения, включая навигацию, должны быть управляемыми, независимо от способностей;
- понятность (Understandability): любая информация и функциональность МП должны быть понятны всем пользователям;
- надежность (robustness): контент МП должен быть достаточно надежным, чтобы его можно было легко интерпретировать различными пользовательскими агентами, включая вспомогательные технологии.

При этом принципы обеспечения доступности, учитывающие специфику МП, находятся в стадии разработки [126,127].

В соответствии с требованиями принятого в Российской Федерации ГОСТа Р 52872-2019 [101] все Интернет-ресурсы, мобильные приложения, цифровые программы и другие пользовательские приложения должны обеспечивать доступность для широкого круга пользователей с ограничениями жизнедеятельности, в том числе нарушениями ментальной сферы. К настоящему времени по данным по данным национальной системы качества России (Роскачество) 95% приложений, выпускаемых на рынок ПО, не адаптированы для людей с ограниченными возможностями здоровья [102].

Анализ исследований по видам доступности МП для разных категорий пользователей с ограничениями жизнедеятельности представлен в приведены в [125] (рисунок 1.6).

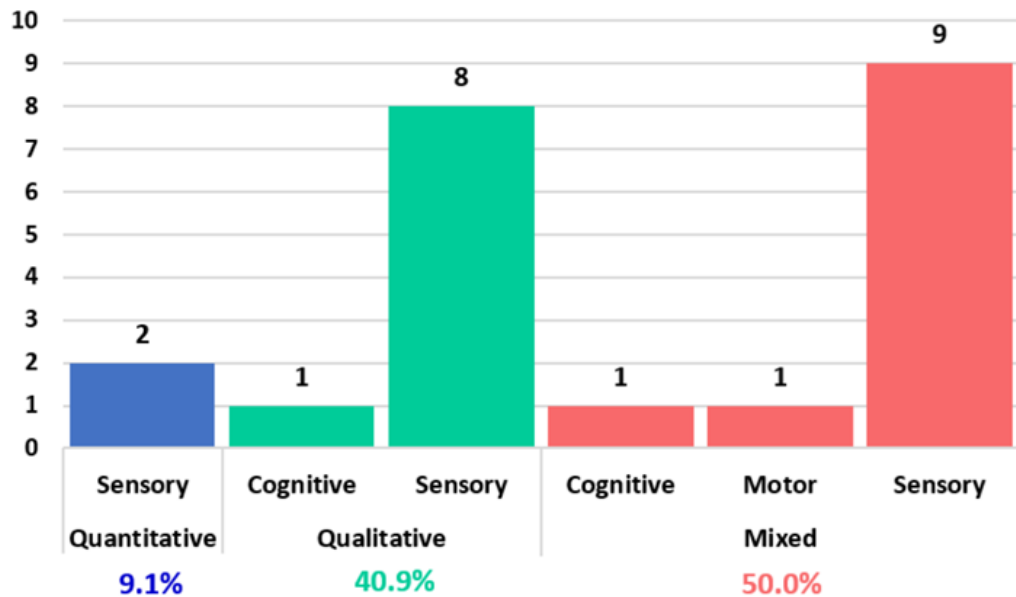


Рисунок 1.6 — Исследования по видам доступности МП

1.4.3 Способы разработки интерфейсов МП для пользователей с ИН

Одним из наиболее эффективных способов повышения доступности ПО является возможность адаптировать функционал и/или интерфейс ПО под конечного пользователя [91-95]. Многие современные МП для людей с ИН предусматривают те или иные возможности адаптации, в большинстве случаев это настройки интерфейса (шрифтов, цветовой палитры, звукового сопровождения и т.д.), которые можно осуществлять на этапе эксплуатации МП [96-100]. Так, в [129] предложена технология проектирования интерфейсов ИП для пользователей с ИН, основанная на типовых решениях, отвечающая определенным требованиям интерфейса к удобству использования. В [130] разработан адаптивный пользовательский интерфейс (AUI) и адаптивные меню, описаны различные стили адаптации и политики адаптации, которые используются в адаптивных меню, а также их преимущества и недостатки. МП для управления повседневной деятельностью, доступного для пользователей с ИН, описанное в [137], позволяет настраивать параметры интерфейса лицам, осуществляющим уход за людьми с ИН. Примеры интерфейса МП приведены на рисунке 1.7.

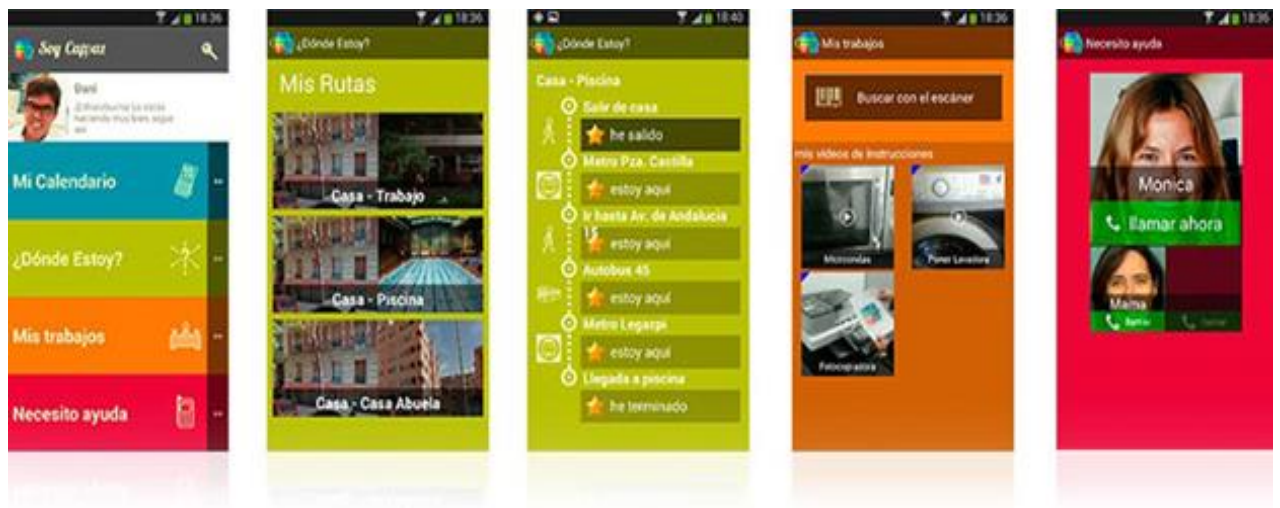


Рисунок 1.7 — Экранные формы Soy Carrpasz Mobile Application

Интерфейс МП “Цифровой помощник”, разработанного для путешествий с для людей с ИН [96], включает набор пиктограмм, разработанных специально для людей с ИН. Примеры экранных форм МП показаны на рисунке 1.8.



Рисунок 1.8 — Экранные формы МП “Цифровой помощник”

Таким образом, в современных МП для людей с ИН доступность обеспечивается либо за счет ad-hoc разработки неизменяемого (non-adaptable) интерфейса, априорно проектируемого с учетом общих принципов доступности, либо разработки адаптируемого (adaptable) интерфейса, апостериорно настраиваемого под конечного пользователя с ИН на этапе эксплуатации ЖЦ МП

(в большинстве случаев это параметрические настройки шрифтов, цветовой палитры, звукового сопровождения и т.д.). Для оценки доступности МП применяют либо методы априорного экспертного оценивания интерфейса МП на соответствие принципам доступности, либо методы внешнего наблюдения за действиями пользователя МП с интерфейсом, уже адаптированным под этого пользователя, и оценки его способности самостоятельного управления МП [11, 12].

Параметрическая адаптация интерфейса в целом позволяет повысить доступность МП. Однако возможности и ограничения людей с ИН сильно различаются. В современных классификациях болезней выделяются до пятидесяти вариантов интеллектуальных нарушений, характеризующихся различными признаками и проявлениями. Для повышения доступности МП для широкого круга пользователей с ИН требуется более глубокая персонализированная, не только параметрическая, но и структурная адаптация интерфейса. Обеспечение такого уровня адаптации интерфейса требует поддержки динамического контента МП, что необходимо предусматривать уже на этапе проектирования ЖЦ МП. Для разработки МП для пользователей с ИН требуется разработка модели ЖЦ, учитывающей особенности требований к обеспечению доступности МП для людей с ИН, и моделей, методов и технологий проектирования МП, применимого в специализированных CASE-средствах сквозной поддержки этапов ЖЦ МП для такой категории пользователей.

1.5 Выводы

В первой главе проведен анализ подходов к разработке МП и рассмотрены существующие технологии и программные средства для поддержки ЖЦ ПО:

- рассмотрено понятие и этапы ЖЦ ПО, включающие анализ требований, проектирование, реализация, тестирование и эксплуатацию;
- описаны программные средства для поддержки ЖЦ ПО, построена сравнительная таблица этапов ЖЦ и соответствующих им программных средств;

- описан ЖЦ МП и его особенности от МП ПО;
- рассмотрены технологии разработки МП и их отличия;
- описаны программные средства для поддержки ЖЦ МП, проведена классификация средств поддержки ЖЦ МП, рассмотрены системы управления разработкой МП, системы управления МП, приведены примеры МП для людей с ИН;
- рассмотрены особенности разработки мобильных приложений для людей с ИН, в т.ч упоминается, что возможности и ограничения людей с ИН сильно различаются, в результате чего требуется более глубокая персонализированная адаптация;
- сформулированы цели и задачи работы.

2. Разработка метода автоматизированного проектирования МП для пользователей с ИН

2.1 Модель ЖЦ МП с адаптируемым интерфейсом

В зависимости от характера и проявления интеллектуальных ограничений потенциальные пользователи МП с ИН могут сталкиваться с различными проблемами, такими как, например, невозможность прочесть текст на экране МП, понять назначение элементов управления и других объектов интерфейса, выбирать и перемещать объекты приложения, управлять приложением и т.д. Для обеспечения доступности для пользователей с различными ИН МП должно предоставлять возможность настройки экранов под конечного пользователя без потери полноценного доступа к функционалу. Поэтому для пользователей с ИН требуется глубокая персонализированная адаптация интерфейса (возможность изменения количества элементов управления, выбора различных пиктограмм для отображения элементов управления, использования или неиспользования текста и т.д.). Такая адаптация интерфейса требует поддержки динамического контента МП, соответственно данные возможности необходимо предусматривать уже на этапе проектирования МП, т.к. это может потребовать изменений в программном коде. Кроме того, многие люди с ИН не смогут самостоятельно настраивать интерфейс, что также нужно принимать во внимание при разработке МП для таких пользователей.

Применение успешных интерфейсных решений, реализованных в МП и апробированных с пользователями с ИН, для разработки новых МП может существенно повысить качество таких МП и повысить уверенность разработчиков МП в их доступности пользователям с ИН. Необходимость учета особенностей пользователей с ИН может также потребовать разработки ряда специальных функций, не зависящих от назначения разрабатываемого МП. Повторное использование кода этих функций позволит упростить процесс разработки и повысить скорость сборки проекта МП.

Таким образом, модель ЖЦ МП должна учитывать специфику пользователей с ИН и обеспечивать возможность адаптации интерфейса под каждого пользователя в соответствии с его возможностями и ограничениями. А средства поддержки разработки МП, обеспечивающие возможность адаптации их интерфейса под конечного пользователя должны объединять типичные возможности систем управления МП на этапах разработки и эксплуатации, а также расширять их за счет добавления функций адаптации интерфейса разработанных МП.

Модель ЖЦ МП для людей с ИН с учетом особенностей такой категории пользователей и специфики процесса разработки МП для таких пользователей, должна включать следующие этапы:

- этап идентификации: анализ проблем и потребностей людей с ИН, разработка концепции МП для пользователей с ИН;
- этап проектирования: разработка проекта МП с повторным использованием готовых проектных решений, проектирование шаблонов экранов МП, формирование наборов элементов управления для настройки интерфейса под конечного пользователя;
- этап реализации: реализация прототипа МП с интеграцией готовых решений, выбранных из хранилища.
- этап прототипирования с тестированием: итерационная разработка прототипа с тестированием с людьми с ИН и их опекунами до получения минимально жизнеспособного продукта (MVP), доступного пользователям с ИН;
- этап развертывания: загрузка МП в соответствующий магазин/рынок приложений для массового пользователя;
- этап эксплуатации: настройка опекуном интерфейса МП под конечного пользователя ИН.

Разработана модель ЖЦ МП, отражающая специфику разработки МП для людей с ИН [116]. Этапы и процедуры ЖЦ МП для людей с ИН приведены на рисунке 2.1 (добавленные этапы и процедуры ЖЦ показаны жирной-линией).

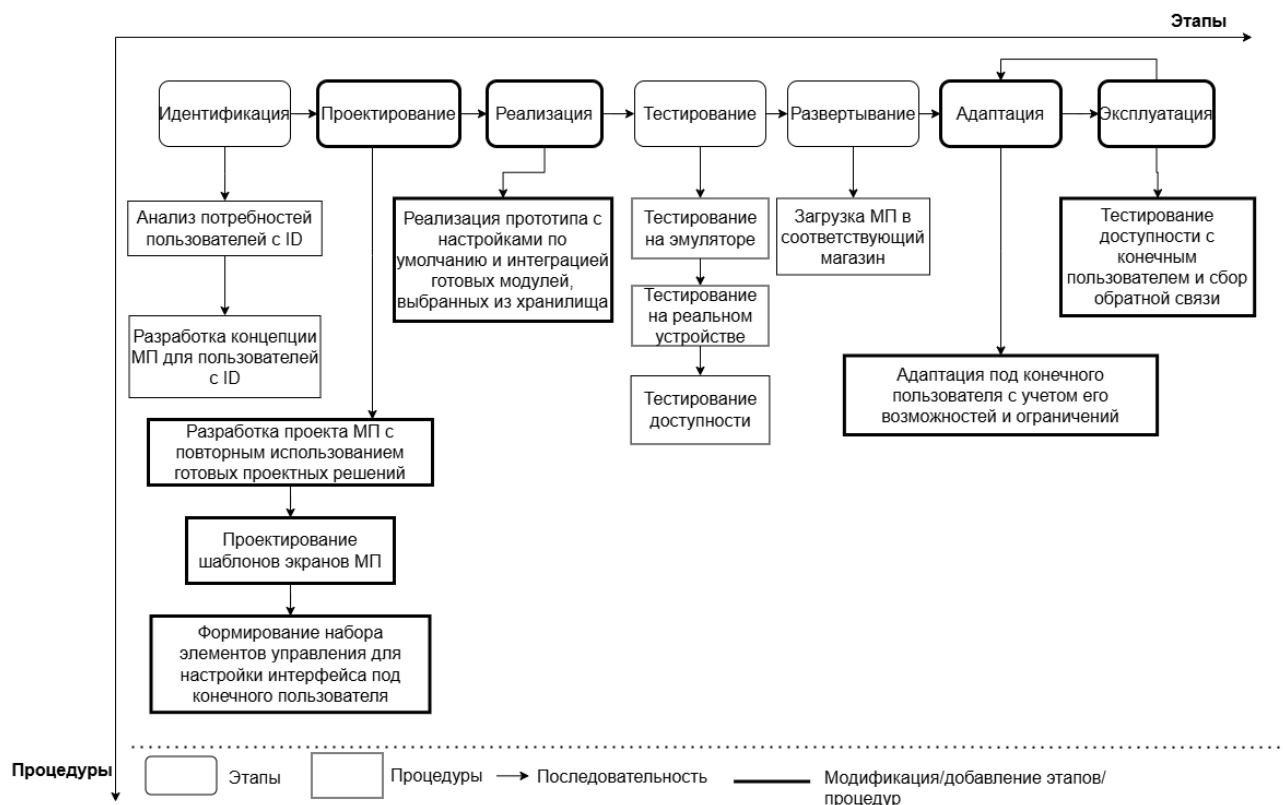


Рисунок 2.1 — Этапы и процедуры ЖЦ МП для людей с ИН

2.2 Концепция сквозной поддержки этапов ЖЦ МП

Для поддержки предложенной модели ЖЦ необходимо обеспечить возможность сквозной разработки МП, включающей этапы создания концепции, проектирования, разработки и развертывания решения. Наличие единой команды, отвечающей за весь процесс разработки, также может помочь упростить общение и принятие решений, т.к. все члены команды участвуют в проекте с самого начала. Это приводит к более эффективному процессу разработки и более качественному конечному продукту [117].

На этапе проектирования МП необходимо предусмотреть возможность повторного использования кода. Для реализации этого требования применена технология модуляризации, т.е. выделения некоего функционала программы в логически законченные, максимально независимые модули, решающие конкретную пользовательскую проблему, с четко обозначенными внешними зависимостями [132, 133], которая адаптирована под разработку МП для пользователей с ИН. Функционал МП разделяется на независимые модули,

которые можно переиспользовать при разработке других МП. Каждый модуль для МП разделен на независимые слои и компоненты, доступные для переиспользования при разработке других модулей.

Для обеспечения возможности адаптировать интерфейс МП на этапе эксплуатации без изменения программного кода необходимо поддерживать не только статические данные, но и динамические - получаемые с сервера.

Процесс сквозной разработки МП с адаптируемым интерфейсом и связи между этапами ЖЦ МП показаны на рисунке 2.2.

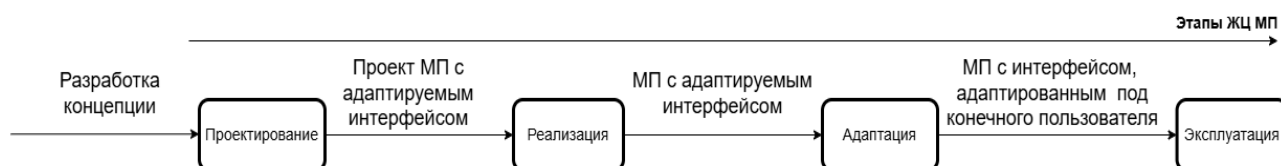


Рисунок 2.2 — Процесс сквозной разработки МП с адаптируемым интерфейсом

2.3 Метод проектирования интерфейса МП

Метод автоматизированного проектирования адаптируемого интерфейса МП для людей с ИН основан на Pattern-Based концепции [13, 136]. Проектирование интерфейса включает разработку шаблонов всех экранов МП, на которых размещаются все возможные элементы интерфейса, требуемые для доступа к функционалу МП, разделение элементов шаблонов на обязательные и опциональные элементы, сопоставление каждому элементу интерфейса набора возможных пиктограмм (изображений) и прикрепление элементов «по месту». Обязательными являются элементы, которые необходимы для работы отдельно взятого МП и должны включаться в интерфейс каждого пользователя. Опциональные элементы могут включаться в интерфейс в зависимости от возможностей и ограничений пользователя. Под общими элементами понимаются элементы, которые содержатся более чем на одной странице. Общие элементы применяются для настройки интерфейса на всех страницах, содержащих эти элементы.

Разработана модель адаптируемого интерфейса МП, представляемая кортежем вида:

$$Interface = \langle CommonElements, Pages \rangle, \quad (1)$$

где *CommonElements* – множество общих элементов страниц; *Pages* – множество шаблонных описаний страниц.

$$Pages = (page_{i1}, \dots, page_{ij}, \dots, page_{im}), \quad (2)$$

где $page_i$ – i -тая страница интерфейса, n – количество страниц.

$$page_i = (element_1, \dots, element_j, \dots, element_m), \quad (3)$$

где $element_j$ – j -тый элемент i -той страницы, m – количество элементов i -той страницы.

$$element_{ij} = \{status(n/o), size, color, type (c/u), Pictograms\}, \quad (4)$$

где $status/(n/o)$ – статус j -того элемента (обязательный/опциональный),

$size$ – размер элемента, $colour$ – цвет элемента,

$type (c/u)$ – тип j -того элемента (общий/уникальный),

Pictograms – множество пиктограмм, сопоставленных j -тому элементу.

Шаблоны интерфейсов, разрабатываемые на этапе проектирования интерфейса, отображают расположение обязательных и опциональных элементов экрана (элементы управления, текстовые поля и т.д) и варианты их возможных изображений. Сплошной линией обозначаются обязательные элементы, пунктирной - опциональные. Пример шаблонного интерфейса страницы МП представлен на рисунке 2.3.

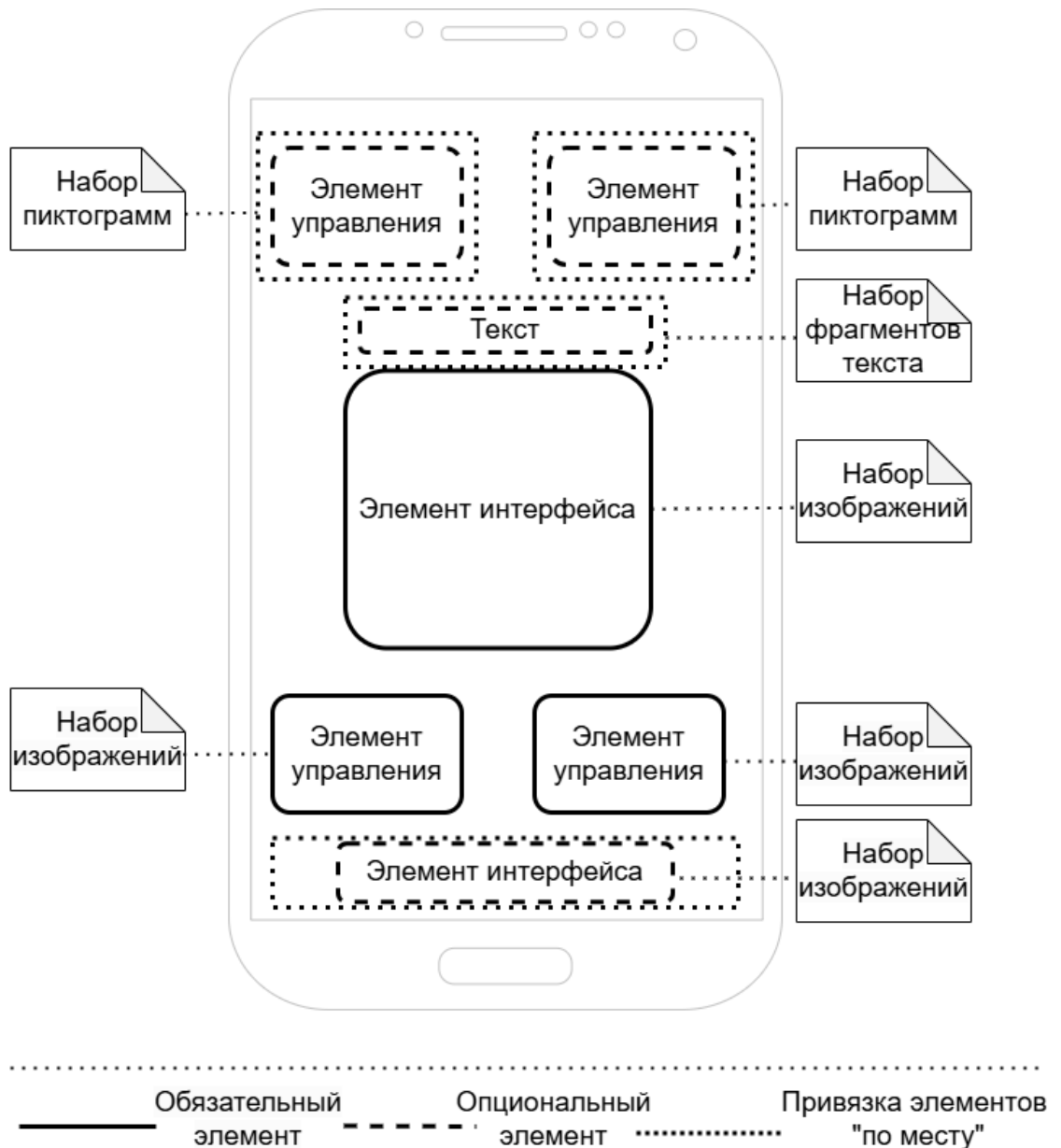


Рисунок 2.3 — Пример шаблонного интерфейса страницы МП

Процесс адаптации интерфейса МП под конкретного пользователя заключается в выборе элементов интерфейса для каждой страницы и их изображений (пиктограммы и/или другие графические иллюстрации) из шаблонных описаний, оптимальном для этого пользователя с учетом его возможностей и ограничений. Процесс адаптации интерфейса ориентирован на опекунов людей с ИН (родители, медицинские работники), знающих возможности и ограничения своих подопечных.

2.4 Шаблон проектирования МП

В настоящее время для разработки МП с повторным использованием кода применяют технологии модуляризации, разработанные различными компаниями-разработчиками МП, такими как «Лаборатория Касперского» [118], Surf [119], Яндекс [120].

Технология модуляризации, применяемая в компании «Лаборатория Касперского», основана на различении двух типов модулей: core- и feature-модулей. В feature-модуле определяется функционал, выделенный непосредственно разработчиком в конкретном программном приложении. В core-модулях определяются части кода, реализующие общие для мобильных приложений функции, такие как утилиты, работа с сетью, работа с БД и т. д., интерфейсы в такие модули не включаются. Core-модуль может быть декомпозирован на отдельные логически законченные части. Названия модулей формируются из двух частей: вначале указывается тип модуля (core или feature), затем добавляется логическое название модуля (scanner, network и т.д.).

Каждый feature- или core-модуль разбивается на два блока: API и Impl (core-impl и feature-impl). В API находится внешний программный интерфейс, через которое можно обращаться к этому feature- или core-модулю. API блок ничего не знает об устройстве своего модуля, это абсолютно изолированный блок. В состав модульной структуры могут также добавляться utils-модули, в которые помещаются функционал, не требующий разделения на api и impl. В модулях feature-impl находится бизнес-логика приложения. Модули группы feature- impl могут иметь доступ к модулям группы API или Utils, но они не имеют к другим модулям группы Impl. Все внешние зависимости feature-модулей аккумулируются в api внешних зависимостей. Пример возможной схемы многомодульного приложения, приведенный в [121], показан на рисунке 2.4.

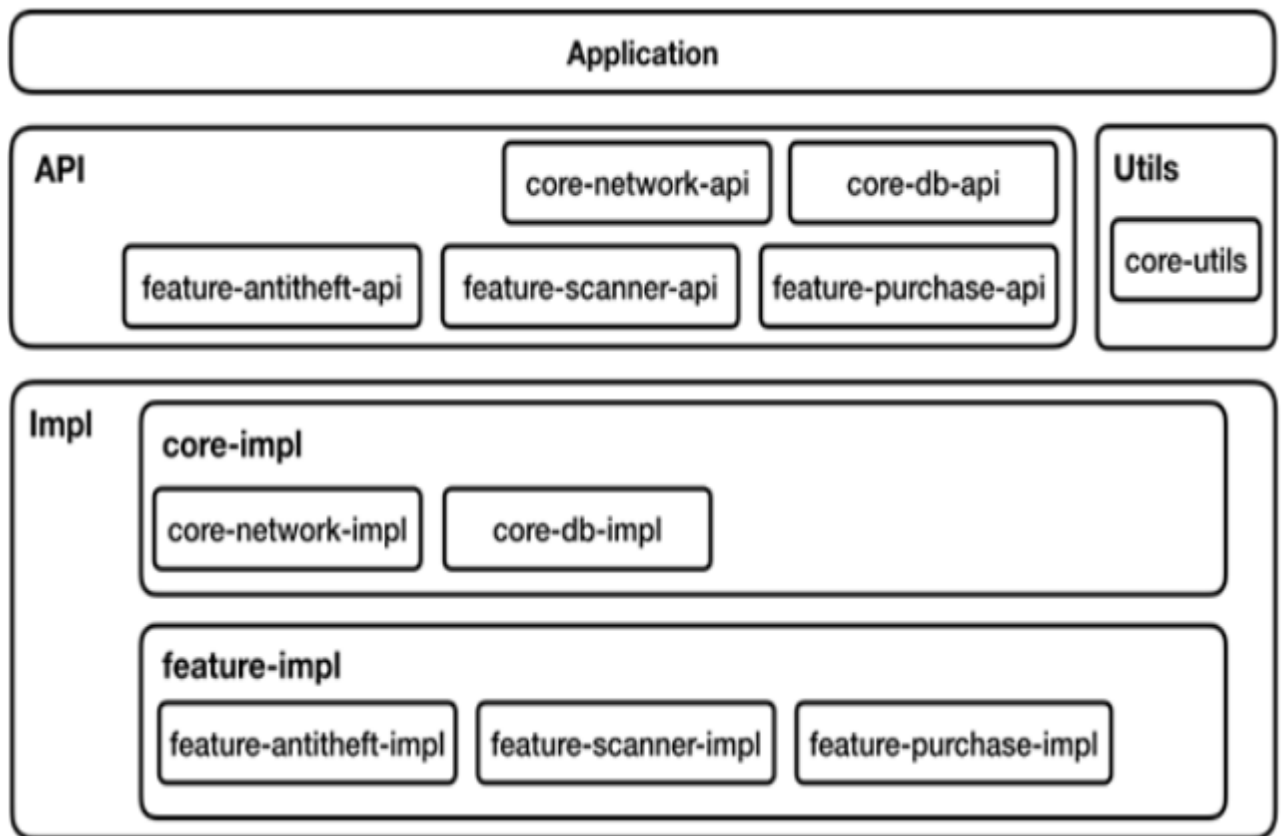


Рисунок 2.4 — Пример схемы многомодульного приложения

Технология разработки многомодульных мобильных приложений, основанной на принципах CleanArchitecture, применяемая в компании Surf, занимающейся разработкой мобильных приложений, описана в [119]. Основной особенностью CleanArchitecture является строгое разделение архитектуры приложения на слои, каждый со своей областью ответственности, и направление зависимостей между слоями снизу вверх (слои верхних уровней не должны зависеть от слоев нижних уровней).

Технология модуляризации, основанная на многослойной архитектуре, подразумевает построение иерархии модулей и привязку модулей к слоям архитектуры. Модуль App-injector является корневым модулем иерархии, он знает обо всех модулях в проекте, но о нем не знает ни один; кодогенерация размещается в корневом модуле.

Внешние зависимости поставляются через Base модуль; модули слоя Interactor могут зависеть друг от друга. Модули более высоких слоев могут зависеть от нескольких более низких, но от них может зависеть только корневой.

Такая технология модуляризации обеспечивает “правильное” соблюдение принадлежности модулей к слоям, при котором модули низших слоев “ничего не знают” о более высоких; скорость сборки проекта, реализующего такую технологию, сильно увеличивается в сравнении с одномодульными проектами, т.к. при изменениях в коде собираются только модули, претерпевшие изменения, модули, зависящие от них, а также корневой модуль; обеспечивает параллелизм сборки модулей, т.к. независимые модули могут собираться параллельно, и возможность переиспользования модулей.

Все модули приложения разделяются на feature- (f-name-) и interactor-(i-name) модули. В feature-модули выносятся разработка конкретного функционала. Это может быть несколько экранов с одной семантикой (например авторизация), может быть один большой экран со сложной логикой. feature-модуль может быть разбит по более мелким пакетам - логическим блокам (например, конкретным экранам, относящимся к одному функционалу), не зависящим друг от друга.

Основным модулем приложения (точкой входа) является App-injector, который “знает” обо всех модулях и является единственным модулем с зависимости предоставляемыми dagger библиотекой. От App-injector не зависит ни один другой модуль, он же зависит от всех. В Domain-модуль выносятся код, относящийся к работе с моделями данных.

В Base-модуле объявляются зависимости, необходимые для использования во всех других модулях (через api), описываются базовые сущности, использующиеся как в i-, так в f-модулях. Base-ui - базовый слой для модулей функционала программы и пользовательских представлений. Base-ui зависит от слоя Base и включает базовые классы данного слоя для всех функционалов, а также переиспользуемые сущности, которые нельзя выделить в отдельный функционал (или же это не имеет смысла). Base-ui включает в себя пакет util, который содержит полезные сущности, хелперы, методы для работы с UI, и также может включать в себя пакет navigation для навигации между модулями. Пример иерархии модулей приведен на рисунке 2.5.

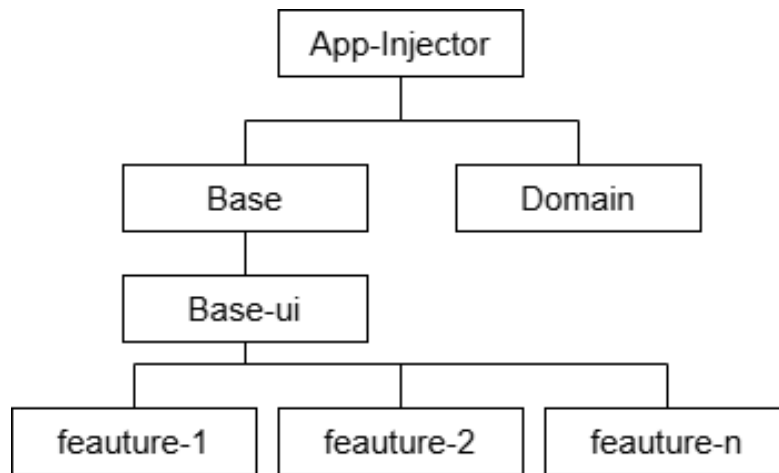


Рисунок 2.5 — Пример иерархии модулей

Проведен анализ применимости технологий модуляризации для разработки МП для людей с ID по следующим критериям:

- процесс переноса модуля в другое приложение должен быть понятным разработчику, не обладающему большим опытом;
- архитектура модулей должна содержать минимальное количество слоев, достаточных для разработки МП для пользователей с ID;
- способ хранения модулей должен позволять хранить все версии релизов модулей, а также историю изменения;
- технология должна описывать архитектуру МП в шаблонах для генерации базового кода.

По совокупности требований к архитектуре МП для пользователей с ИН наиболее подходящей по выбранным критериям является технология, разработанная компанией Surf. Разработана многомодульная архитектура МП для пользователей с ИН как модификация архитектуры Surf: слойное деление на модули заменено на деления по слоям в самом модуле, что позволило сократить количество модулей, но оставить слойное деление; количество типа модулей сокращено до двух из-за того в Surf использовалось слойное деление на уровне модулей. Модифицированная таким образом архитектура разделяет приложение на три уровня модулей: модуль приложения (App), функциональные модули (Feature), библиотечные модули (Library) (рисунок 2.6).

Технология разработки МП реализует принципы модуляризации [122] применительно к МП для пользователей с ИН.

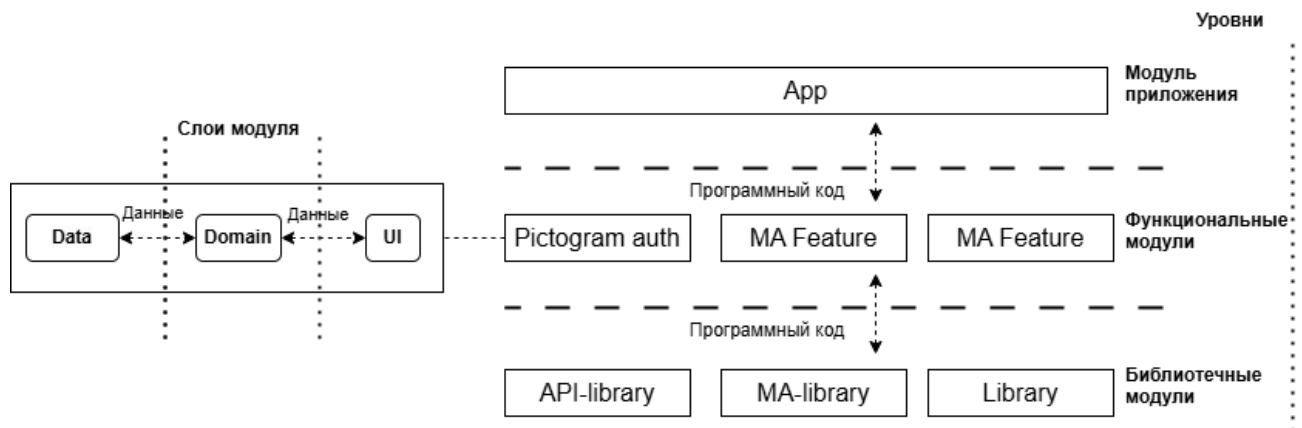


Рисунок 2.6 — Многомодульная шаблонная архитектура МП

Модуль приложения связывает вместе функциональные модули. Функциональные модули соответствуют полноэкранным, согласованным функциональным возможностям пользователя в приложении. Модуль приложения имеет внутреннюю архитектуру, которая состоит из трёх слоев. В первом слое располагаются источники данных и запросы к ним, например, запросы к API или базе данных (БД). Во втором слое содержится бизнес логика приложения. В третьем слое располагаются компоненты, необходимые для визуализации интерфейса. В библиотечных модулях функциональность разделена между несколькими модулями. Библиотеки могут зависеть друг от друга, они предоставляют общий функционал, который повторно используется в нескольких функциях.

App-модуль является точкой входа в приложение. Он зависит от функциональных модулей (feature-модулей) и обеспечивают корневую навигацию.

Feature-модули представляют собой изолированную часть функциональности разрабатываемого МП, которая соответствует экрану или набору тесно связанных экранов, например, авторизация пользователя с помощью пиктограмм.

К библиотечным (library-модулям) относятся модули, предназначенные для реализации функционала, который можно повторно использовать для разработки других МП. Разработка МП для пользователей с ИН на основе шаблонной архитектуры позволяет разработчикам использовать готовые решения, применяемые в предыдущих проектах, и добавлять новые решения как модули для повторного использования.

Также модуль имеет внутреннюю архитектуру, которая состоит из трех слоев. В слое Data располагаются источники данных и запросы к ним, например запросы к API или к базе данных. В слое Domain создается такая сущность как Interactor, которая содержит в себе бизнес логику приложения. В слое UI располагаются компоненты необходимы для отрисовки интерфейса, в случае android это activity, fragments или views.

В библиотечных модулях функциональность разделена между несколькими модулями. Разные библиотеки могут зависеть друг от друга. Библиотеки предоставляют общий функционал, который повторно используется в нескольких или во всех функциях.

Разработка МП для людей с ИН на основе разработанной технологии модуляризации позволяет разработчикам МП использовать готовые решения, успешно реализованные в предыдущих своих и чужих проектах, и добавлять новые успешные решения как модули, применяемые для повторного использования. При этом разработчик МП не тратит время на разработку архитектуры МП, а пользуются уже готовым шаблон для генерация базового кода экрана.

Таким образом, работа с функционалом шаблона проектирования МП для пользователей с ИН включает следующие этапы:

1. скачивание примера с шаблонами генерации базовых классов для экрана и примером многомодульной архитектурой;
2. разработка архитектуры МП на основе выбранного шаблона;
3. поиск готовых модулей в хранилище (например, модуля для работы с данными пользователя через API);

4. подключение готовых модулей в проект МП;
5. выбор модулей для выгрузки в репозиторий.

Метод автоматизированного проектирования МП с адаптируемым интерфейсом включает следующие шаги:

1. Разработать концепцию МП для пользователей с ИН.
2. Спроектировать функционал МП, необходимый для реализации концепции МП.
3. Разработать шаблоны экранных форм интерфейса МП и сопоставить каждому элементу набор пиктограмм для его отображения на экране.
4. Создать проект МП и разработать архитектуру МП на основе шаблонной архитектуры.
5. Выбрать готовые модули из хранилища модулей, реализующие требуемые функции МП и подключить их к проекту МП
6. Спроектировать остальной функционал МП, требуемый в рамках концепции МП.
7. Выбрать из проекта МП модули, применяемые для повторного использования, и добавить их в хранилище моделей.
8. Реализовать MVP (Minimum Viable Product) МП с элементами интерфейса и пиктограммами для их отображения, выбранными по умолчанию разработчиком.

2.5 Выводы

Во второй главе разработан метод автоматизированной разработки мобильных приложений для людей с ограниченными интеллектуальными возможностями и рассмотрены следующие концепции и технологии:

- разработана модель ЖЦ МП для пользователей с ИН, описана специфика пользователей, этапы и процедуры модели;
- предложена концепция сквозной поддержки этапов ЖЦ МП для различных этапах поддержки и эксплуатации МП;

- разработан шаблон проектирования интерфейса МП, основанная на Pattern-Based концепции;
- разработана модель адаптируемого интерфейса МП, представленная кортежем;
- предложен метод автоматизированного проектирования МП.

3 Разработка CASE-системы поддержки ЖЦ МП для пользователей с ИН

3.1 Архитектура системы

Для поддержки жизненного цикла (ЖЦ) ПО применяют CASE инструменты и методы, используемые на разных этапах ЖЦ ПО для анализа, проектирования, поддержки разработки, тестирования и сопровождения ПО.

Разработана концепция специализированной CASE-системы, реализующей предложенный метод автоматизированной разработки МП для пользователей с ИН. Система объединяет типичные возможности MADM и MAM систем, расширяет их за счет добавления функций, обеспечивающих возможность адаптации интерфейса пользователя МП (рисунок 3.1).

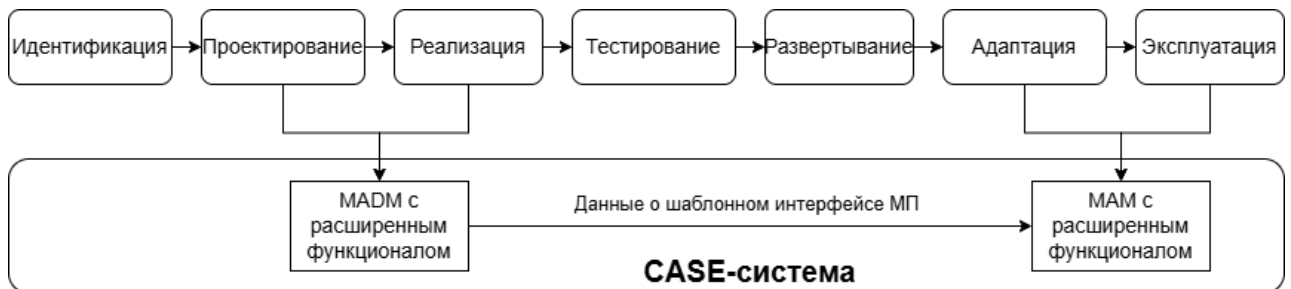


Рисунок 3.1 — Концепция CASE-системы

Разработана архитектура CASE-системы, обеспечивающая поддержку сквозной разработки МП. В CASE-системе предусмотрены отдельные компоненты под каждую сервис-ориентированную аудиторию. Данные компоненты системы обеспечивают независимое развитие различных направлений в поддержке ЖЦ МП, а также гарантируют кросс-взаимодействие на различных этапах поддержки и эксплуатации МП.

CASE-система, реализующая предложенный метод автоматизированного проектирования МП для пользователей с ИН, включает подсистемы поддержки этапов проектирования, реализации, адаптации и эксплуатации МП, базу данных настроек интерфейса и ссылок на добавленные в хранилище модули МП, API для взаимодействия системы с МП. Архитектура системы в нотации ArchiMate приведена на рисунке 3.2.

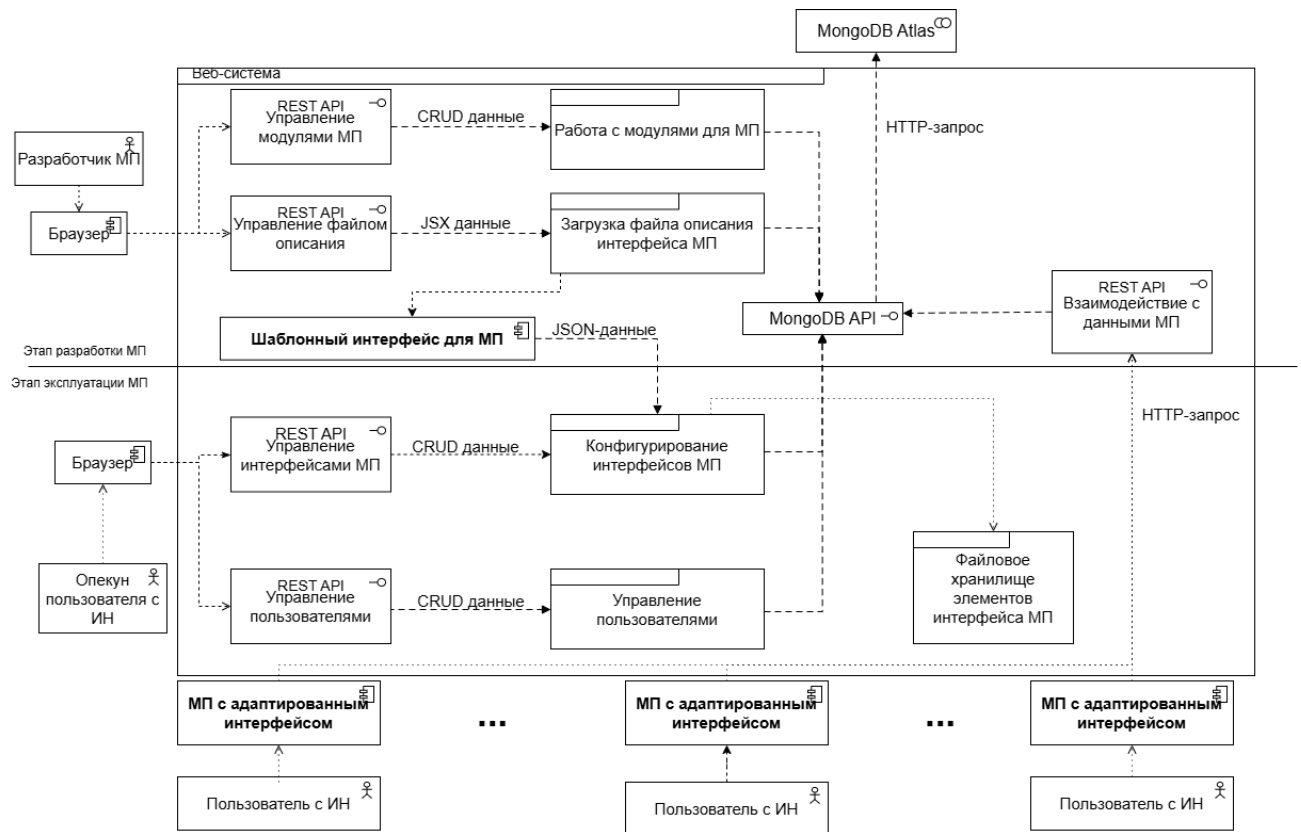


Рисунок 3.2 — Архитектура системы (нотация ArchiMate)

Система построена на REST архитектурном стиле, который устанавливает однозначное соответствие между операциями, используемыми при работе с персистентными хранилищами данных, и HTTP-методами, и обеспечивает таким образом независимость способа подключения МП к БД.

Система реализована как веб-приложение. Для бэкенд-разработки выбрана платформа Node.js, для фронтенд-разработки применены HTML, CSS, JavaScript. В качестве БД использована MongoDB. Для обмена данными между сервером и МП использован протокол HTTP, для поддержки разработки МП использованы специализированные наборы заранее разработанных компонентов и библиотек, реализована поддержка наследования и ограничение доступа к определенным компонентам системы. Детальная таблица стека технологий системы представлена в таблице 3.1.

Таблица 3.1 — Стек технологий системы

Платформа	Веб	
Backend	Node.js	
Frontend	HTML, CSS, JavaScript, React	
База данных	MongoDB	
Протокол обмена	HTTP	
Библиотеки	Express	фреймворк веб-приложений для Node.js
	EJS	язык шаблонов, позволяющий генерировать HTML-разметку
	Mongoose	связь с БД MongoDB
	Passport	совместимое с Express промежуточное ПО аутентификации для Node.js
	React	JavaScript-библиотека (для КП)

3.2 Функционал системы

Основной функционал системы включает создание проекта МП на основе многомодульной шаблонной архитектуры, просмотр хранилища модулей, выбор и подключение готовых модулей в проект разрабатываемого МП, добавление файла описания шаблонов страниц МП, добавление модулей в

хранилище модулей, конфигурирование интерфейса МП под конкретного пользователя на этапе эксплуатации с помощью специального веб-интерфейса, встраиваемого на этапе разработки МП. Этапы ЖЦ МП для пользователей с ИН, поддерживаемые CASE-системой, показаны на рисунке 3.3.

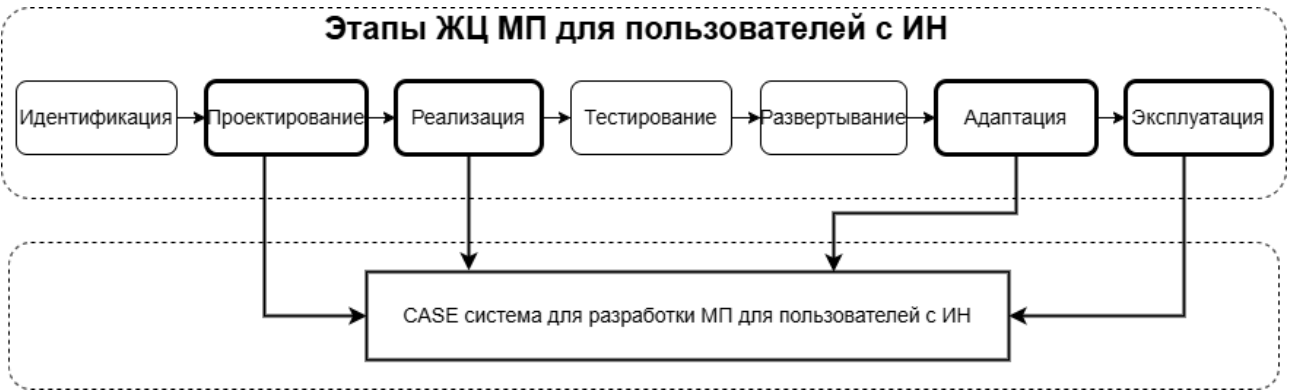


Рисунок 3.3 — Этапы ЖЦ МП для пользователей с ИН, поддерживаемые CASE-системой

Процесс работы пользователей (опекун, Разработчик МП) в системе представлен на рисунке 3.4.

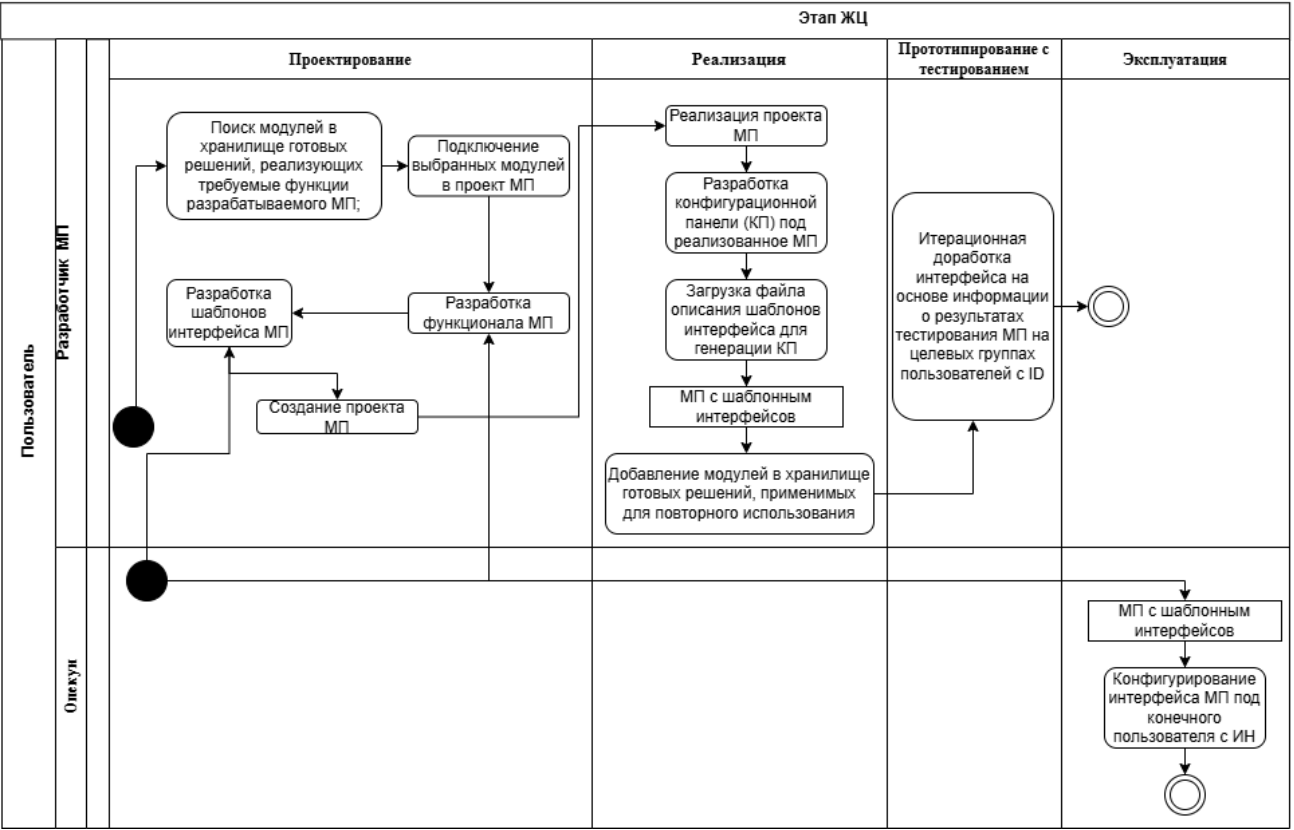


Рисунок 3.4 — Процесс работы пользователей (опекун, Разработчик МП) в системе (activity diagram)

3.3 Конфигурационная панель для адаптации интерфейса МП

Разработка адаптируемого интерфейса МП для людей с ИН основана на концепции шаблонных интерфейсов [74]. Процесс проектирования интерфейса включает разработку шаблонов всех экранов МП, на которых размещаются все элементы интерфейса, требуемые для доступа к функционалу МП, и разделение элементов шаблонов на обязательные и опциональные элементы.

Модель интерфейса МП, реализуется как JSON-объект [123], полями которого являются страницы интерфейса МП. Поле содержит информацию о параметрах конфигурации элемента: обязательность/необязательность, размер и цвет элемента, принадлежность к разряду общих элементов. Под общими элементами в модели представления понимаются элементы, которые содержатся более чем на одной странице. Общие элементы необходимы для возможности настройки элемента интерфейса на всех страницах, на которых он представлен. Имеется возможность настраивать повторяющиеся элементы без привязки к общим, настраивая элемент на каждой странице, где он представлен. Поле, описывающее общие элементы, содержится в корне модели представления, как и страницы МП.

В системе реализована проверка правильности составления описания структуры файла. При обнаружении синтаксической ошибки в файле в процессе генерации интерфейса МП предусмотрен вывод сообщения о характере ошибке и рекомендации по ее устранению.

Для разработки веб-интерфейса, позволяющего конфигурировать интерфейс МП на этапе эксплуатации, требуется программировать backend и frontend логику веб-интерфейса под каждое разрабатываемое МП. Такой способ разработки веб-интерфейса каждый раз требует привлечения разработчика веб-системы, что существенно ограничивает возможности применения CASE-системы. Для исключения разработчика веб-системы из процесса разработки веб-интерфейса предложен метод адаптации интерфейса с помощью КП, которая

создается разработчиком МП и встраивается в CASE-систему для каждого МП на этапе реализации [131].

Разработана технология адаптации интерфейса МП на основе шаблонов и способ адаптации интерфейса с помощью КП, встраиваемой в систему управления МП [116].

КП представляет собой функционал, генерирующий веб-интерфейс, и включает два экрана: экран для ввода или копирования кода для шаблонного интерфейса МП, и экран визуализации конфигурируемого интерфейса. Встраиваемый в систему веб-интерфейс позволяет конфигурировать интерфейс МП под конечных пользователей без изменения программного кода. Пример шаблона КП представлен на рисунке 3.5.

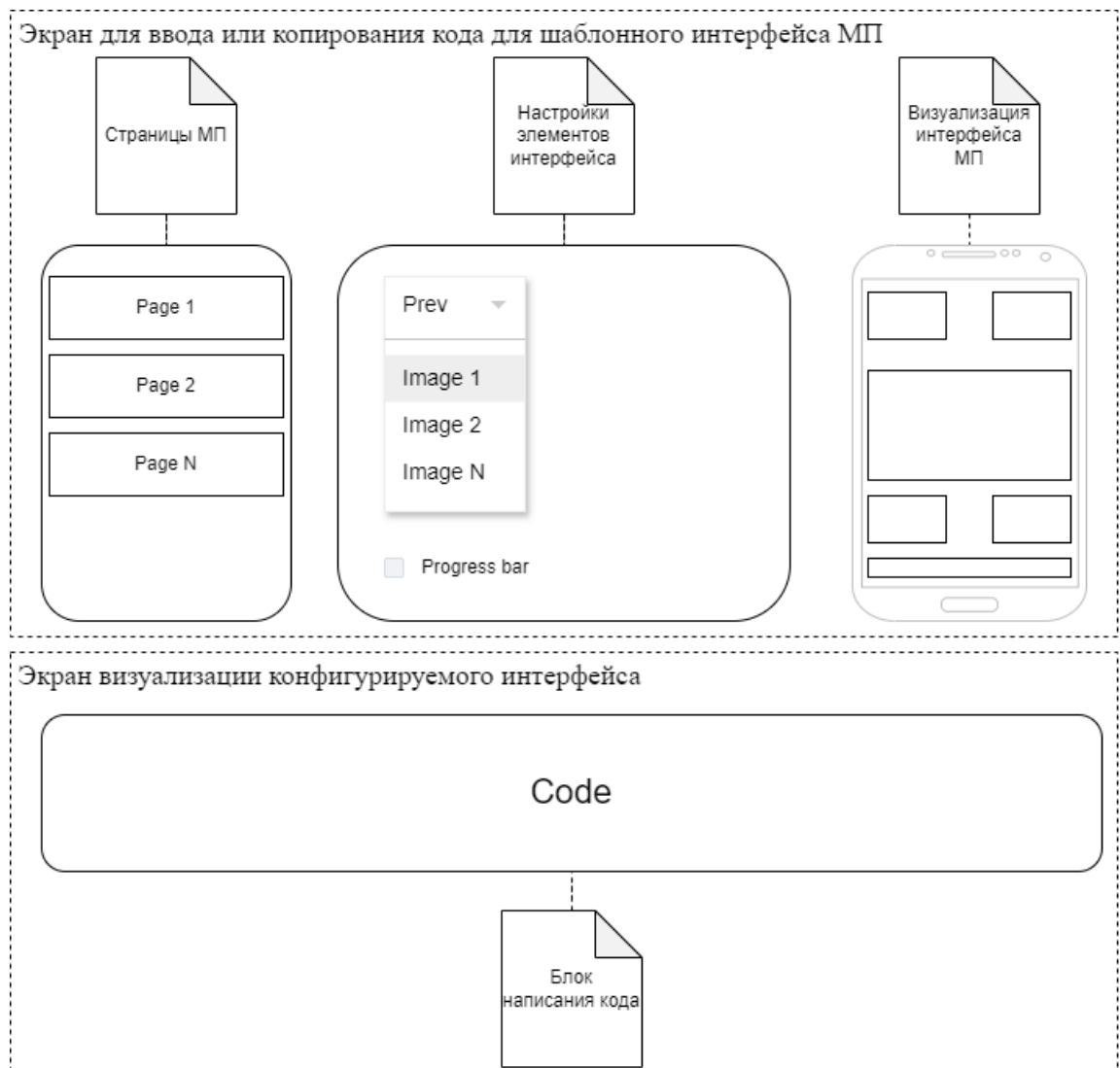


Рисунок 3.5 — Пример шаблона КП

Процесс генерации КП строится на валидации, парсинге описания интерфейса МП и построении блока элементов управления и блока элементов представления на основе тегов и информации, содержащейся в них. Редактор, также являющийся частью разработанного модуля, позволяет предварительно сгенерировать КП на основе составленного описания и проверить файл описания на наличие ошибок.

Процесс генерации КП включает следующие этапы:

- валидация;
- парсинг;
- рендеринг блока представления интерфейса МП.

Рендеринг включает этапы:

- инициализация МП;
- инициализация страниц МП;
- инициализация элементов интерфейса МП;
- рендеринг блока элементов управления.

КП включает область настроек, которая содержит виджеты для настройки элементов интерфейса, конфигурируемого МП и область, в которой отображается текущая страница интерфейса МП.

Настройка интерфейса включает выбор элементов интерфейса для отображения и их изображений на экране МП (пиктограммы и/или другие графические иллюстрации).

Разработанный способ адаптации интерфейса с помощью КП не требует изменения исходного кода МП и может осуществляться лицом, не владеющим навыками разработки МП.

В процессе валидации описание проверяется на соответствие требованиям. В случае, если описание составлено не по требованиям, в результате генерации пользователю будет выведено сообщение, объясняющее, в чем заключается ошибка в описании интерфейса МП.

Процесс генерации содержит парсинг, необходимый для преобразования описания интерфейса МП в данные, которые будут использованы при рендеринге КП.

Рендеринг блока представления начинается с инициализации МП. После завершения инициализации генератор получает команду на рендеринг страниц приложения. Каждая страница приложения генерируется отдельно. В процессе рендеринга, страница сообщает генератору об инициализации с именем, указанным в атрибуте тега страницы. Генератор сохраняет информацию о странице в модели представления интерфейса.

Далее генератор рендерит дочерние элементы страницы - настраиваемые и не настраиваемые элементы интерфейса МП. В процессе рендеринга каждый настраиваемый элемент интерфейса сообщает генератору об инициализации, передавая имя, указанное в атрибуте элемента, настраиваемые параметрами (например размер, цвет) и имя родительской страницы. Генератор сохраняет информацию в модели представления интерфейса.

После того, как процесс рендеринга закончен, на основе модели представления интерфейса, генератор рендерит блок элементов управления. Генерация происходит постранично. Так как у элементов интерфейса может быть несколько настраиваемых параметров, страница делится на блоки, отвечающие за разные элементы интерфейса. Каждый блок содержит название настраиваемого элемента интерфейса. Каждый элемент в блоке содержит название настраиваемого параметра элемента интерфейса.

Алгоритм генерации КП приведен на рисунке 3.6.

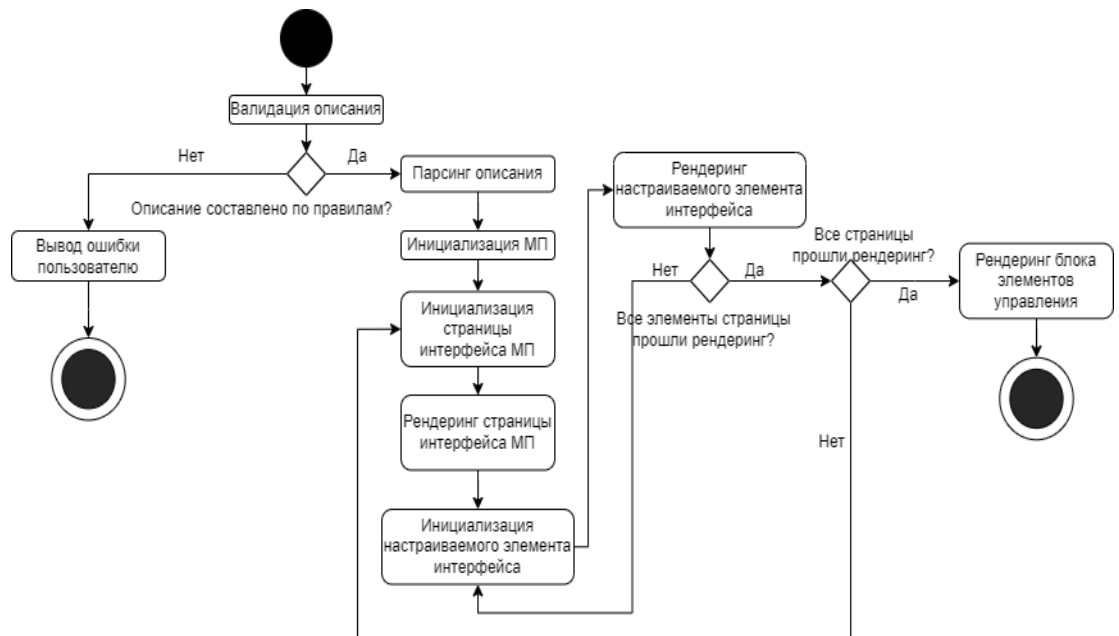


Рисунок 3.6 — Алгоритм генерации КП (Activity diagram, UML)

Процесс генерации интерфейсов МП с помощью КП включает следующие действия: добавление приложения в систему (выполняется разработчиком МП) с названием и описанием приложения, а также файлом, описывающим интерфейс; описание интерфейса должно выполняться с применением JSX - расширения языка JavaScript; редактирование загруженного приложения; просмотр списка приложений в системе; удаление приложения. Процесс генерации интерфейсов МП с помощью КП приведен на рисунке 3.7.

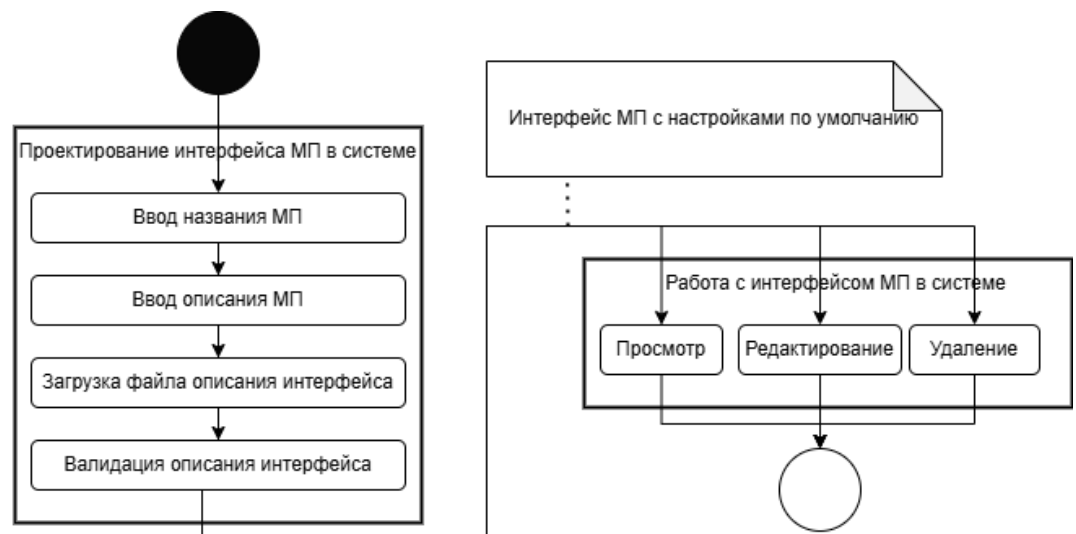


Рисунок 3.7 — Процесс генерации интерфейсов МП с помощью КП (Activity diagram, UML)

3.4 Способ реализации многомодульной архитектуры МП

Подсистема работы с модулями МП предоставляет возможности для разработки МП на основе многомодульной архитектуры, позволяя выбирать готовые функциональные решения из хранилища модулей. Для создания базового кода функционального модуля применяются шаблоны. В качестве шаблонизатора используется FreeMarker [134], который подходит для написания сложных шаблонов.

Реализация МП в системе основана на многомодульной архитектуре, разработанной на этапе проектирования МП. Для генерации базового кода Feature модуля используются шаблоны (в терминах Android Studio это файл или набор файлов с расширением.ftl, содержащий конструкции на Java или Kotlin и XML. Шаблонизатором является FreeMarker, подходящим для написания сложных шаблонов (рисунок 3.8).

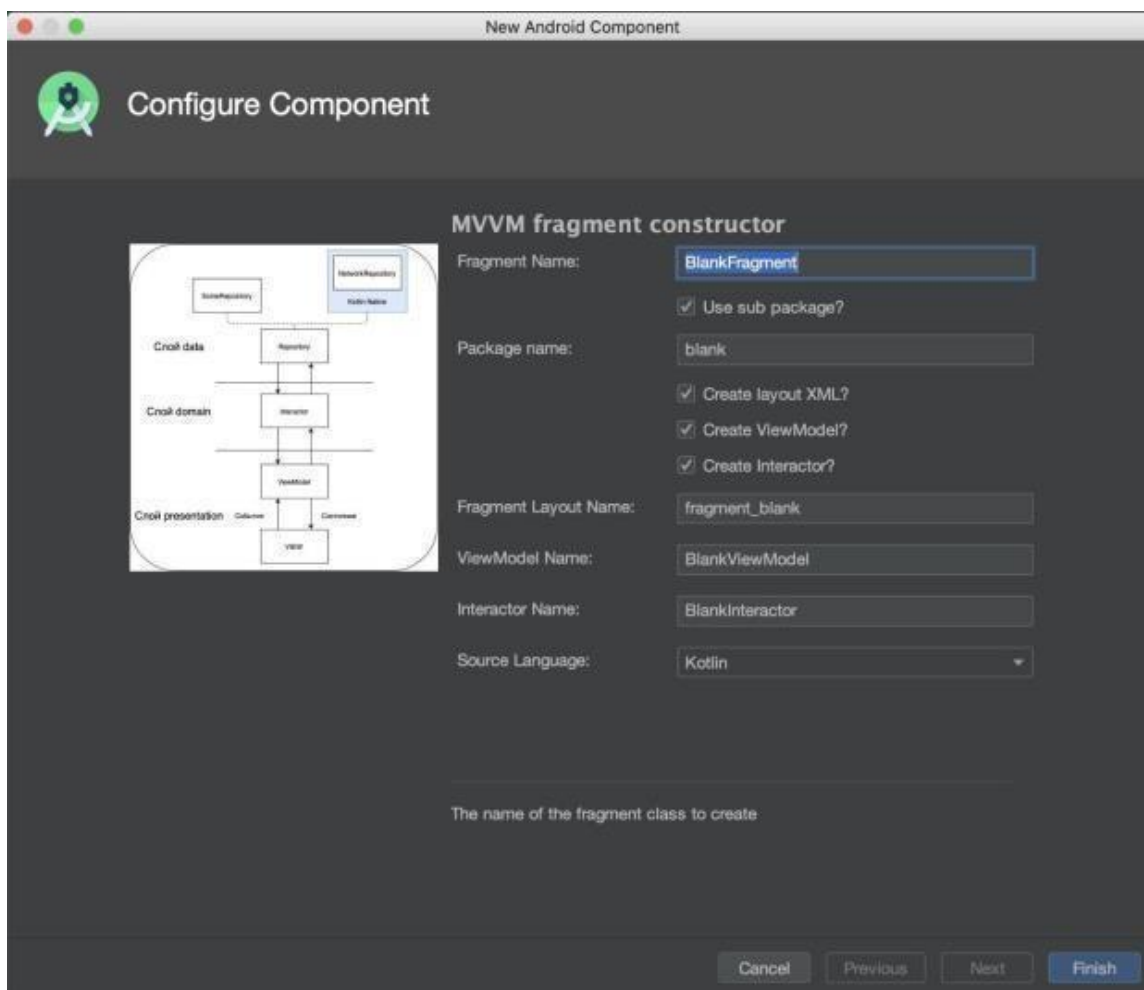


Рисунок 3.8 — Окно настройки и генерации кода по шаблонам

Модульная разработка МП для пользователей с ИН включает этапы: выбор примера с шаблонами генерации базовых классов для экрана; разработка архитектуры МП на основе выбранного шаблона; поиск готовых модулей в хранилище; подключение готовых модулей в МП; выбор модулей для выгрузки в репозиторий.

Разработка МП для пользователей с ИН на основе модуляризации позволяет разработчикам использовать готовые решения, применяемые в предыдущих проектах, и добавлять новые решения как модули для повторного использования.

Разработан способ хранения функциональных модулей. Модули имеют структурированное описание в файле README.md. Способ хранения модулей, включает в себя загрузку модуля разработчиком МП на GitHub и пометку версии модуля тегом (напр. Release Tag v1.0). Затем с помощью сервиса JitPack (репозиторий пакетов для проектов JVM и Android) происходит использование и распространение модулей, находящиеся в Git-репозиториях посредством Gradle (инструмента автоматизации сборки, позволяющий разработчикам МП управлять зависимостями, компиляцией, тестированием и другими аспектами разработки ПО). Способ хранения модулей представлен на рисунке 3.9.

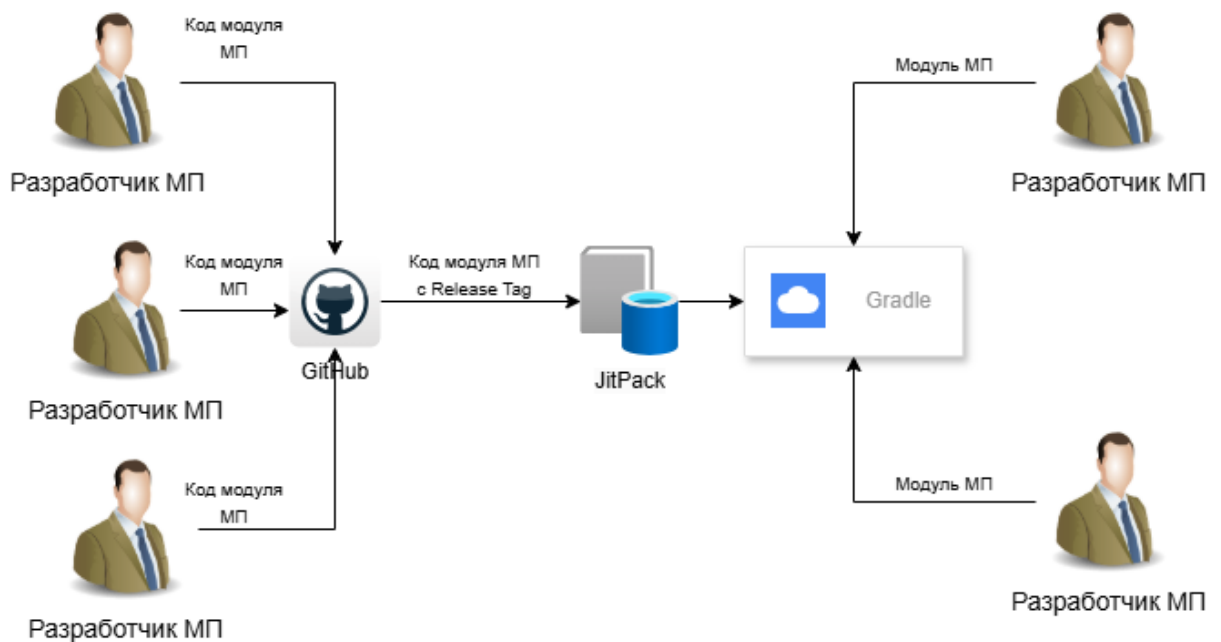


Рисунок 3.9 — Способ хранения модулей

Разработан алгоритм сквозной разработки МП. Входными данными для алгоритма являются дизайн макет с шаблонами страниц МП, созданный разработчиком МП на основе концепции. На этапе проектирования генерируется шаблонный интерфейс МП с настройками по умолчанию. На этапе реализации создается полноценное МП с настройками интерфейса, сделанными разработчиком МП. На этапе адаптации лицо, не владеющее навыками программирования, адаптировать интерфейс конкретного МП под конкретного пользователя с ИН. Алгоритм сквозной разработки МП приведен на рисунке 3.10.

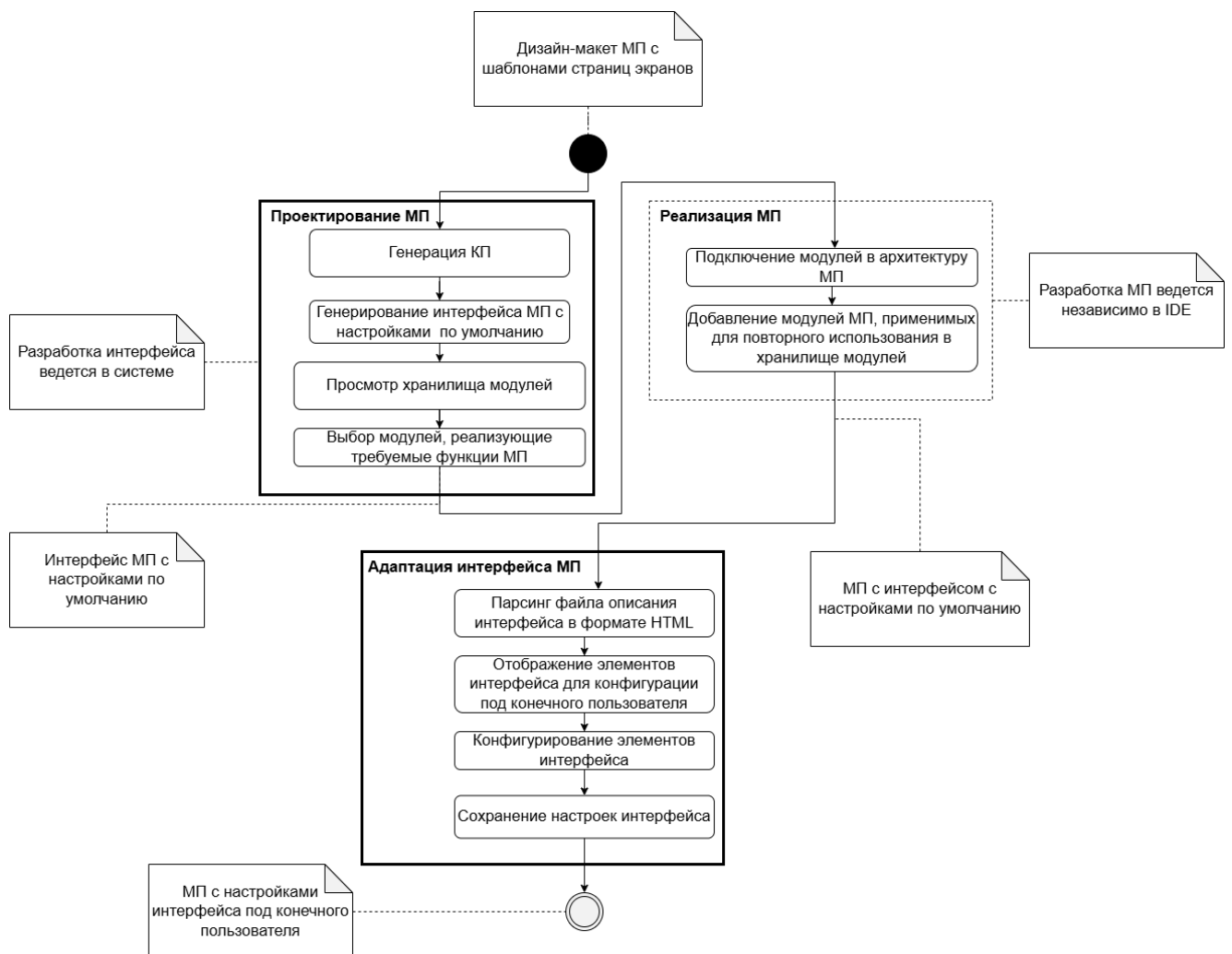
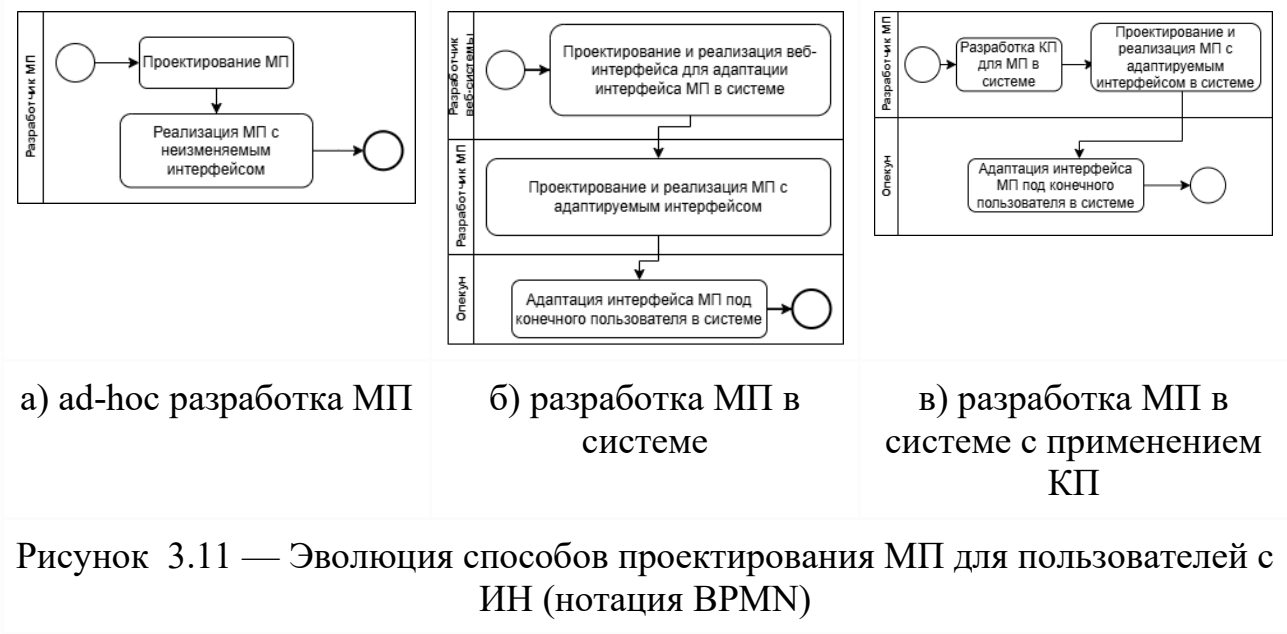


Рисунок 3.10 — Алгоритм сквозной разработки МП (Activity diagram, UML)

Таким образом, разработанный метод автоматизированного проектирования интерфейсов МП, обеспечивает возможность структурной и

параметрической адаптации интерфейсов под конечного пользователя на этапе эксплуатации с помощью веб-интерфейса, встраиваемого в систему. Способ разработка веб-интерфейса в виде конфигурационной панели (КП), позволяет исключить из процесса разработки веб-интерфейса разработчика веб-системы. Эволюция способов проектирования МП для пользователей с ИН показана на рисунке 3.11.



3.5 Интерфейс системы

Разработан интерфейс системы, включающий страницы работы с модулями МП, КП для МП, пользователями системы, адаптацией интерфейсов под конечных пользователей.

Экранная форма списка модулей, поиска форма добавления модуля представлена на рисунке 3.12.

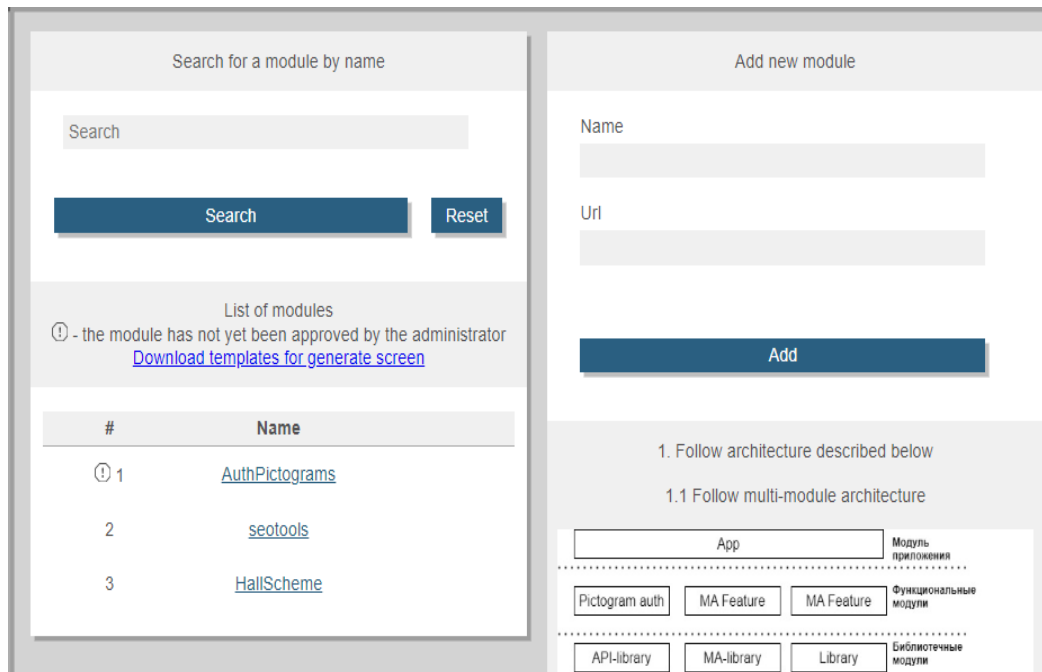


Рисунок 3.12 — Экранная форма списка модулей, поиска и добавления модулей

Экранная форма web-страницы с описанием и добавлением модуля в собственный проект МП представлена на рисунке 3.13.

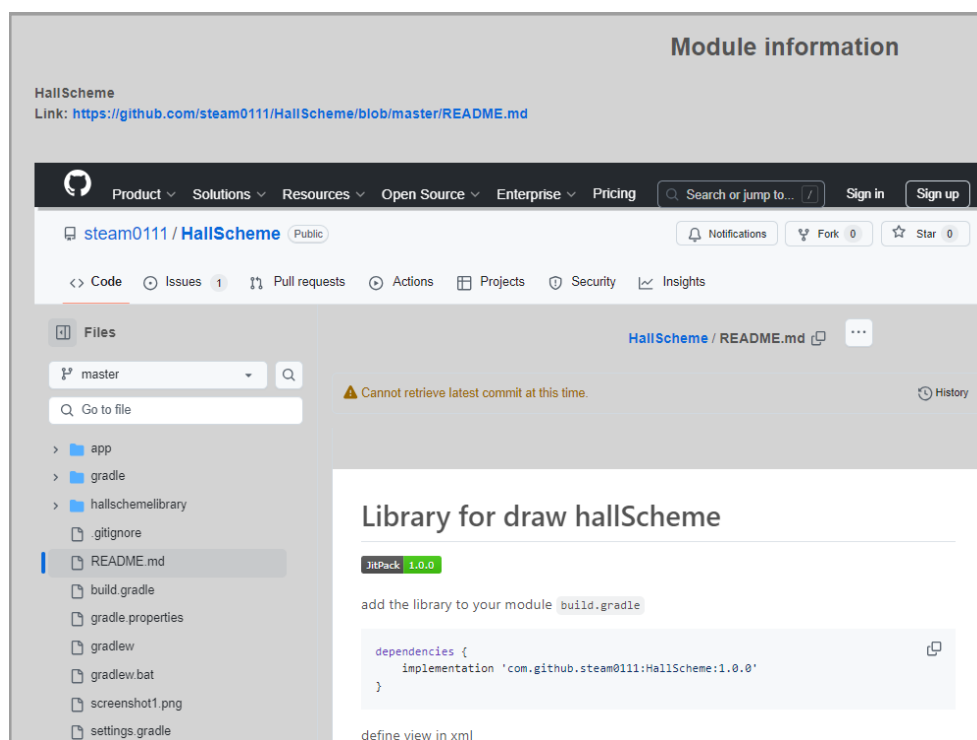


Рисунок 3.13 — Экранная форма с описанием и добавлением модуля в собственный проект МП

Экранная форма web-страницы с выбором и добавлением КП представлена на рисунке 3.14.



Рисунок 3.14 — Экранная форма выбора и добавления КП

Экранная форма web-страницы с генерацией КП представлен на рисунке 3.15.

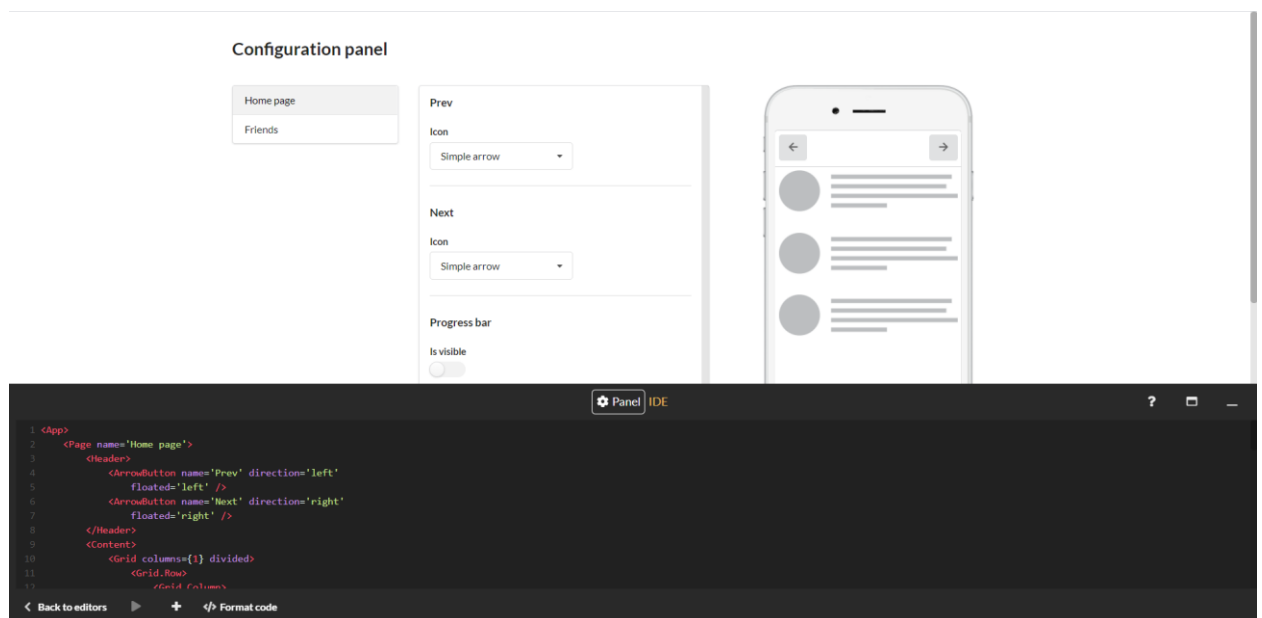


Рисунок 3.15 — Экранная форма для генерации КП

Экранная форма web-страницы просмотра и добавления пользователей представлена на рисунке 3.16.

The screenshot displays two main sections: 'My students' and 'Create new student'.

My students: A table listing three students.

#	Name	Gender	Age
1	John	Male	22
2	Daria	Female	24
3	xxx_iluha_xxx	Male	20

Create new student: A form with two rows of dot patterns for login and password.

Login: Four input boxes with a red 'X' button to the right.

Password: Four input boxes with a red 'X' button to the right.

Below each row of input boxes is a 2x4 grid of dot patterns (1-8) for selection. The patterns are as follows:

- 1: 1 dot (top-left)
- 2: 2 dots (top-left, top-right)
- 3: 3 dots (top-left, top-right, middle-right)
- 4: 4 dots (top-left, top-right, middle-right, bottom-right)
- 5: 5 dots (top-left, top-right, middle-right, bottom-right, bottom-left)
- 6: 6 dots (top-left, top-right, middle-right, bottom-right, bottom-left, middle-left)
- 7: 7 dots (top-left, top-right, middle-right, bottom-right, bottom-left, middle-left, top-middle)
- 8: 8 dots (top-left, top-right, middle-right, bottom-right, bottom-left, middle-left, top-middle, middle-bottom)

Рисунок 3.16 — Экранная форма просмотра и добавления пользователей

Экранная форма web-страницы детального просмотра и редактирования пользователя с выбор МП под конфигурирование представлен на рисунке 3.17.

The screenshot displays two main sections: 'Information' and 'Applications'.

Information: A section for user details.

Profile: A placeholder for a user profile picture and a button labeled 'Выберите файл' (Choose file) with a status 'Файл ...выбран' (File ...selected).

Login: Four input boxes with a red 'X' button to the right.

Password: Four input boxes with a red 'X' button to the right.

Applications: A list of applications for configuration.

- MoneyGame
- MoneyGameSECOND
- AIT
- Third

Configuration Panels: A section with a button labeled 'Open'.

Рисунок 3.17 — Экранная форма детального просмотра и редактирования пользователя с выбор МП под конфигурирование

Экранная форма web-страницы конфигурирования интерфейса МП под конечного пользователя представлен на рисунке 3.18.

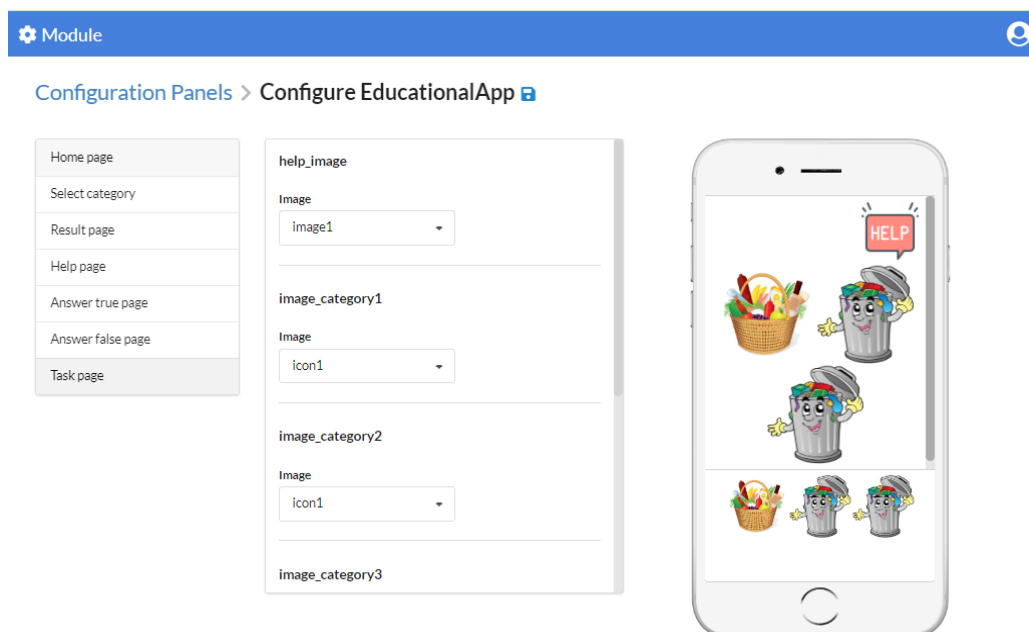


Рисунок 3.18 — Экранная форма web-страницы конфигурирования интерфейса МП под конечного пользователя

Система была размещена на облачной платформе Amvera Cloud [135].

3.6 Методика разработки МП с применением системы

В системе предусмотрены три категории пользователей: администратор, разработчик МП и пользователь, который адаптирует интерфейс МП под конечного пользователя МП с ИН. Таким пользователем может быть родитель, сотрудник мед.центра, опекун, т.е. человек (настройщик интерфейсов), который знает возможности и ограничения своего подопечного и может выбрать интерфейсные элементы и пиктограммы для их отображения, обеспечивающие доступность МП для этого пользователя. Роли пользователей CASE-системы и их действия на этапах ЖЦ МП показаны на рисунке 3.18. Действия на этапах ЖЦ МП показаны в таблице 3.2.

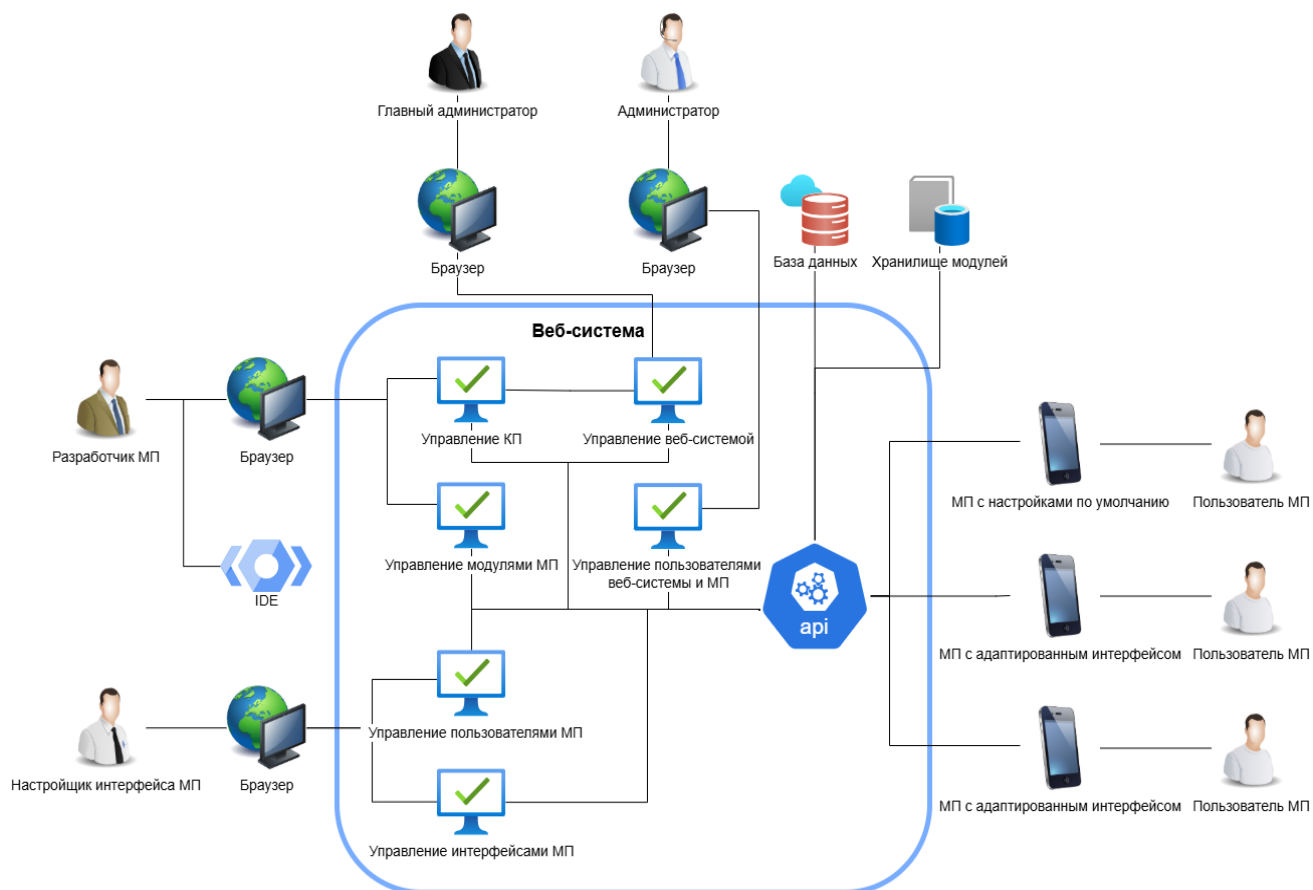


Рисунок 3.18 — Роли пользователей CASE-системы

Таблица 3.2 — Действия пользователей на этапах ЖЦ МП

Этап ЖЦ	Категория пользователя	Действия пользователя в системе
Проектирование	Разработчик МП	Разработка шаблонов интерфейса МП
	Настройщик интерфейса МП	Разработка конфигурационной панели (КП) под МП Создание проекта МП Поиск модулей в хранилище готовых решений, реализующих требуемые функции разрабатываемого МП
Реализация	Разработчик МП	Подключение выбранных модулей в проект МП; (Реализация проекта МП, выполняется в сторонней IDE)

Тестирование	Разработчик МП	Итерационная доработка интерфейса на основе информации о результатах тестирования МП на целевых группах пользователей с ИН
Адаптация	Настройщик интерфейса МП	Конфигурирование интерфейса МП под конечного пользователя с ИН
Эксплуатация	Настройщик интерфейса МП Разработчик МП Администратор	Добавление новых пользователей в систему Добавление модулей в хранилище готовых решений, применимых для повторного использования Управление пользователями

Разработана методика разработки МП для пользователей с ИН с применением системы:

1. разработка концепции МП;
2. разработка дизайн-макетов шаблонов экранов МП;
3. выбор пиктограмм для всех элементов шаблонов экранов МП;
4. создание КП (генерация в системе JSX-файла описания шаблонов страниц МП);
5. создание проекта МП на основе шаблонной архитектуры;
6. выбор из хранилища готовых модулей подходящих решений и добавление в проект;
7. проектирование feature- и library-модулей, реализующих функционал игры и добавление в проект;
8. реализация минимально жизнеспособного продукта (minimum viable product, MVP) МП - реализация полноценного рабочего МП с интеграций с API CASE-системы с шаблонным интерфейсом со значениям по умолчанию (включены все обязательные опциональные элементы интерфейса).

Разработана методика адаптации интерфейса МП под конечного пользователя:

1. выбрать пользователя МП из списка людей с ИН, опекаемых настройщиком интерфейса;
2. выбрать МП из списка МП для адаптации интерфейса под конечного пользователя;
3. открыть КП для выбранного МП;
4. выбрать для каждой страницы МП опциональные элементы управления;
5. выбрать для каждого элемента управления для каждой страницы МП визуальное отображение.

3.7 Онтология проектирования МП для пользователей с ИН

Предложенный в работе подход к разработке МП для пользователей с ИН рассматривает такие приложения как отдельную категорию МП, особенности требований к которым определяют специфику их разработки. Категоризация МП для пользователей с ИН позволяет объединить знания о МП для пользователей с ИН как объекте разработки, о проектных, технологических решениях и средствах, разработанных в рамках предложенного подхода, и описать их как самостоятельную область знаний, что обеспечивает структурированный подход к управлению знаниями и процессами, связанными с созданием МП. Онтология проектирования МП для пользователей с ИН представлена на рисунке 3.19.

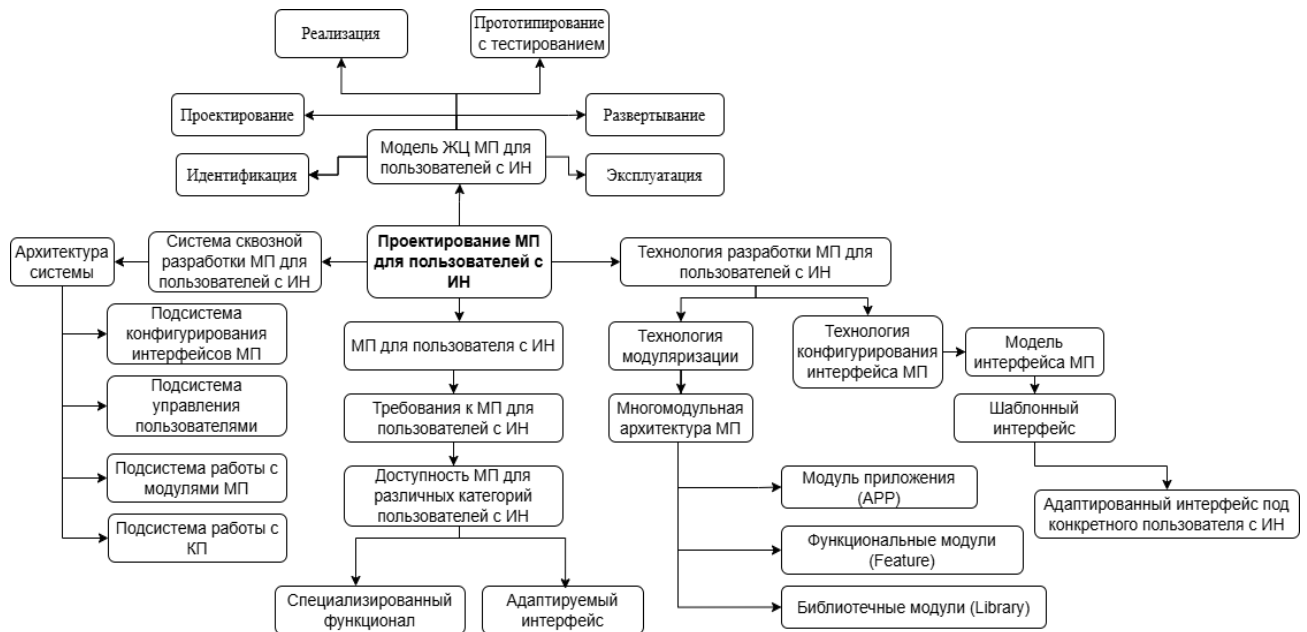


Рисунок 3.19 — Онтология проектирования МП для пользователей с ИН

3.8 Выводы

В третьей главе разработана CASE-система поддержки ЖЦ МП для пользователей с ИН и представлены следующие концептуальные и технические аспекты:

- описана концепция CASE-системы, объединяющая типичные возможности MADM и MAM систем и расширяющая их за счет добавления функций, обеспечивающих возможность адаптации интерфейса пользователя МП;
- реализована архитектура системы, обеспечивающая поддержку сквозной разработки МП и включающая подсистемы поддержки этапов проектирования, реализации, адаптации и эксплуатации МП, базу данных настроек интерфейса и ссылок на добавленные в хранилище модули МП, API для взаимодействия системы с МП ;
- реализован функционал системы и показан процесс работы пользователей;
- разработана КП для системы, основанная на концепции шаблонных интерфейсов;

- представлен шаблон проектирования МП, основанная на технологии модуляризации;
- разработан GUI системы;
- описана методика разработки МП с помощью системы;
- описана методика адаптации интерфейса МП с помощью КП;
- разработана ментальная карта процесса разработки МП для пользователей с ИН, обеспечивающая структурированный подход к управлению знаниями и процессами, связанными с созданием МП;

4. Апробация системы

Разработанная система апробирована на всех группах пользователей. Процесс апробации системы включал в себя:

1. процесс разработки МП с адаптируемым интерфейсом;
2. процесс адаптации интерфейсов разработанных МП под конечных пользователей с ИН;
3. оценку доступности МП с адаптированным интерфейсом каждому пользователю.

Соответственно, в апробации системы принимали участие разработчики МП, потенциальные пользователи системы и пользователи МП (люди с ИН).

В качестве разработчиков МП были привлечены студенты кафедры САПР и ПК Волгоградского государственного технического университета. Разработка МП проводилась в рамках выполнения студентами курсовых проектов по дисциплинам, связанных с проектированием и реализацией ПО, и выпускных квалификационных работ бакалавра (ВКРБ).

Тестирование разработанных МП было организовано в сотрудничестве с коллективами Волгоградского психоневрологического интерната [138] и медицинского центра MPI Oosterlo, Бельгия [139], специализирующимся на работе с людьми с ИН. Волгоградский психоневрологический интернат (Волгоград, Россия) - учреждение, обеспечивающее социальное обслуживание пожилых людей и инвалидов, страдающих хроническими психическими заболеваниями. В учреждении введена в действие программ, направленная на максимально возможную адаптацию людей с инвалидностью к самостоятельной жизни. MPI Oosterlo (Оостерло, Бельгия) - организация, предоставляющая различные виды помощи и поддержки детям, молодым людям и взрослым с ИН.

По результатам апробации системы произведена оценка качества МП для пользователей с ИН и временных затрат на разработку МП в системе.

4.1 Апробация системы на разработке МП с адаптируемым интерфейсом

Апробация системы проводилась на разработке нескольких МП с адаптируемым интерфейсом, предназначенных для пользователей с ИН:

- МП «AIT Test» для выявления предпочтений пользователей с ИН, проявивших интерес к трудовой деятельности, и выбора для них подходящих рабочих мест;
- мобильная игра «Money Game» для тренировки навыков совершения покупок для людей с ИН;
- МП для различения предметов для пользователей с с тяжелыми множественными нарушениями развития (ТМНР).

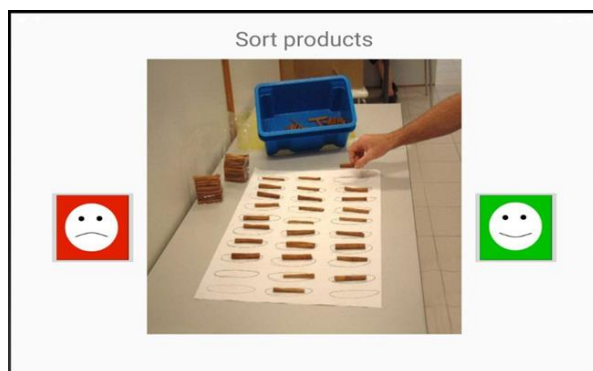
4.1.1 Разработка МП для тестирования предпочтений пользователей

МП “AIT Test” реализует онлайн-тест для анализа предпочтений к различным профессиям, разработанный специально для людей с ИН. Результаты тестирования предназначены для сотрудников специализированных медицинских центров, занимающихся проблемой поиска работы клиентов центра (людей с ИН). Тест представляет собой набор изображений, иллюстрирующих различные профессии, и сгруппированных в категории. Каждой категории сопоставляется соответствующий набор изображений, иллюстрирующий эту категорию. Процесс тестирования заключается в оценивании тестируемым каждого изображения с точки зрения его привлекательности (Нравится/Не нравится) с последующим вычислением результатов тестирования и ранжированием категорий по привлекательности.

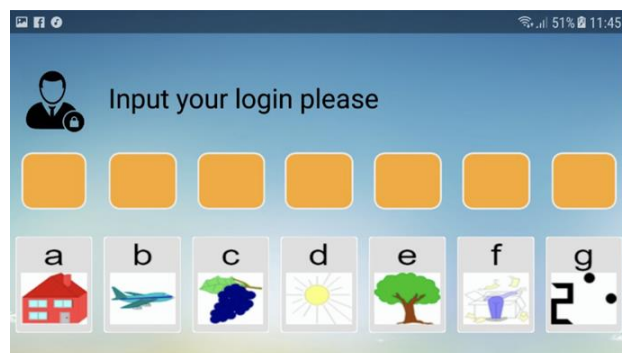
МП реализовано на языке программирования Java. Для разработки МП применена IDE Android Studio.

Первая версия МП была разработана как приложение с неизменяемым интерфейсом. Для разработки пользовательского интерфейса приложения использовалась технологии ориентированного на пользователя проектирования

(User-centered Design) итеративного дизайна (Iterative design). Экранные формы приведены на рисунке 4.1.



а) Экран выбора



б) Экран авторизации

Рисунок 4.1 — Экранные формы первой версии приложения

Тестирование было проведено в MPI Oosterlo [139]. Результаты тестирования приведены в Приложении А.

Полнофункциональная версия МП разработана с применением системы и имеет адаптируемый интерфейс. Экранные формы приведены на рисунке 4.2

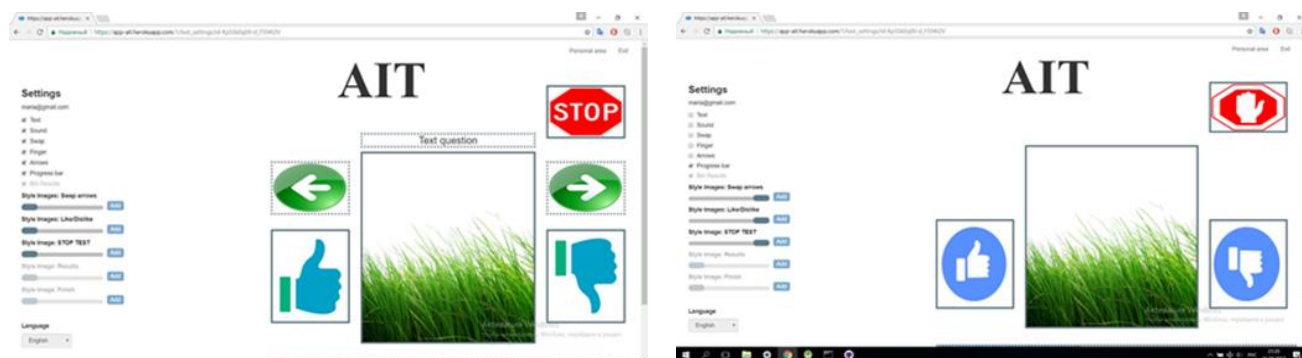


Рисунок 4.2 — Варианты адаптации интерфейса МП AIT Test в системе

4.1.2 Разработка МП для тренировки навыков совершения покупок

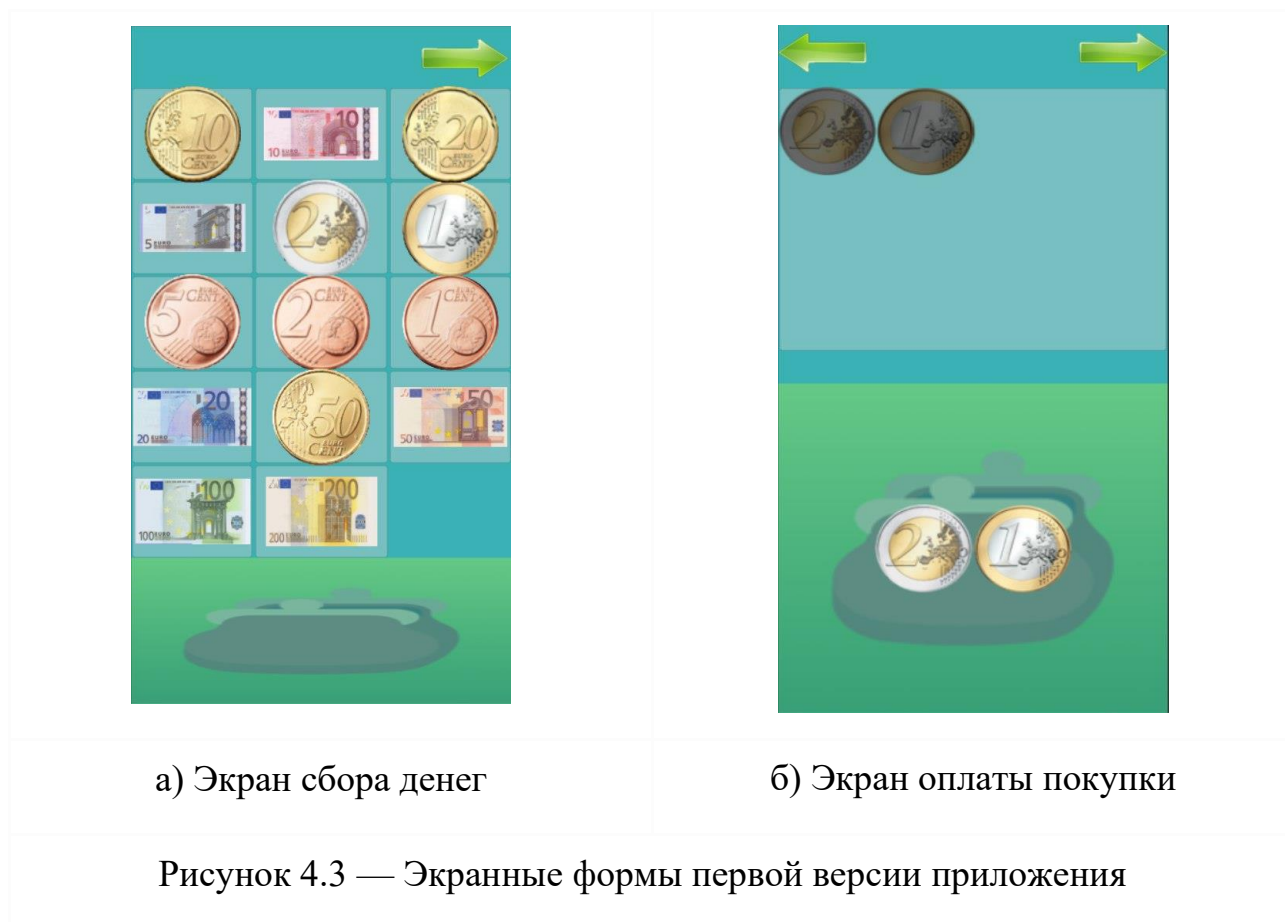
МП “Money Game” предназначена для тренировки навыков совершения покупок для людей с ИН. Функционал МП реализует поддержку следующих этапов процесса совершения покупок:

- управления деньгами (положить в кошелек наличные деньги и/или взять с собой банковскую карту);
- выбор магазина для покупок;

- составление списка покупок в этом магазина;
- поиск товара в магазине;
- оплата товара.

МП реализовано на языке программирования C#. Для разработки МП применена IDE Unity.

Первая версия МП была разработана как приложение с неизменяемым интерфейсом. Экранные формы приведены на рисунке 4.3.



В ходе тестирования были выявлены недостатки разработанного интерфейса:

- шаблоны монет отображались на экране слишком темными, поэтому пользователи плохо различали деньги между собой, путали монеты в 5 и 50 центов;
- изображения товаров оказались слишком маленькими, пользователи не могли разобрать, какие товары представлены для покупки;

- пользователи не всегда понимали, какие объекты, размещенные на экране, нужно выбрать и куда их нужно перемещать для совершения покупки.

Во второй версии приложения эти недостатки были исправлены. Был изменен цвет денег для лучшей различимости элементов, увеличены изображения товаров в виртуальном магазине, для помощи в выборе игровых действий добавлены подсказки. Экранные формы второй версии приложения показаны на рисунке 4.4.



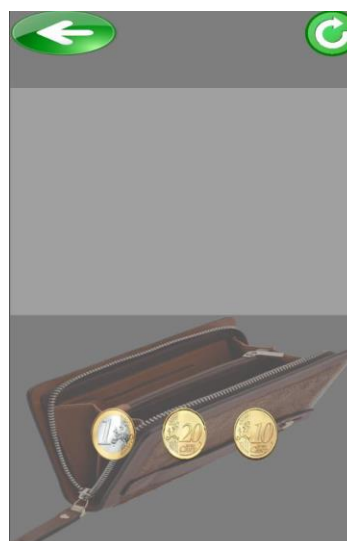
а) Экран сбора денег



б) Экран выбора покупок



в) Экран оплаты покупок



г) Экран получения сдачи

Рисунок 4.4 — Экранные формы второй версии приложения

Процесс тестирования МП с клиентами MPI Oosterlo показан на рисунке 4.5.



а) Имитация сбора денег



б) Имитация сбора денег



в) Тестирование сбора денег



г) Тестирование сбора денег

Рисунок 4.5 — Процесс тестирования МП с клиентами MPI Oosterlo

Третья версия МП была разработана в системе и имела адаптируемый интерфейс. Проект МП был разработан на основе многомодульной архитектуры. Для самостоятельной авторизации пользователей в проект МП был включен разработанный в рамках проекта AIT Test модуль авторизации пользователя с применением пиктограмм, выбранный из хранилища моделей (рисунок 4.6).



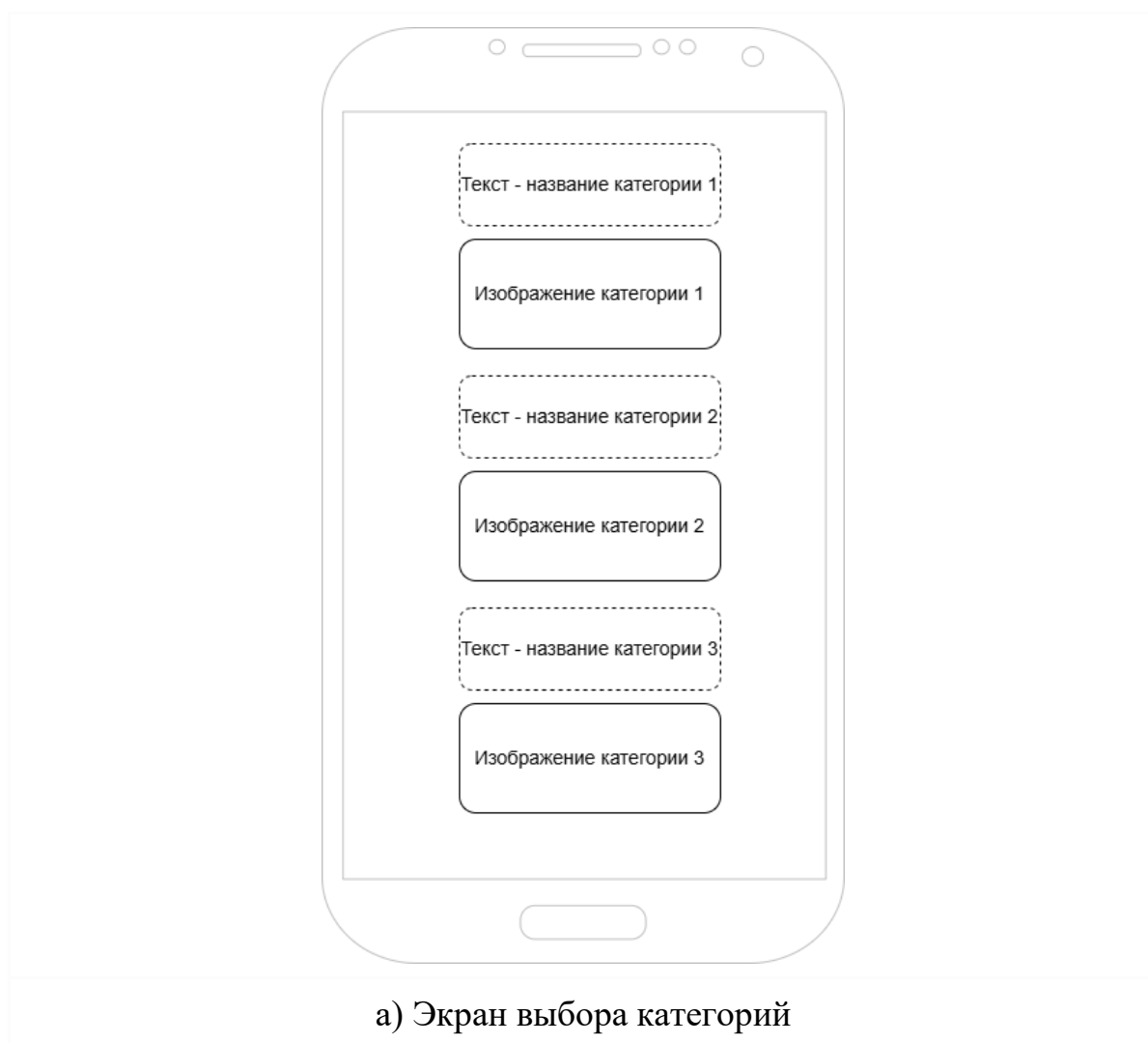
4.1.3 Разработка МП для тренировки навыков различения предметов

Мобильная игра с адаптируемым интерфейсом для пользователей с ТМНР была разработана с применением системы. По Международной классификации болезней (МКБ 10) люди с ТМНР имеют IQ от 20 до 34, в зрелом возрасте их умственное развитие соответствует развитию в возрасте 3-6 лет. Самостоятельная жизнь таких людей практически невозможна; они нуждаются в постоянной заботе и поддержке в повседневной жизни. Однако при определенных условиях люди с ТМНР способны развивать элементарные социальные и коммуникативные навыки. Процесс разработки игры включал следующие шаги:

1. Разработка концепции игры. Игра предназначена для развития навыков различения предметов. Определены категории предметов, окружающих людей с ТМНР в их повседневной жизни. Каждой категории сопоставлены наборы картинок, отображающих эти категории, и предметов, относящиеся к этим категориям. Игровой процесс заключается в

перетаскивании картинки с изображением предмета и наложение на картинку соответствующей категории, к которой принадлежит этот предмет.

2. Проектирование функционала МП. Функционал игры включает функции авторизации с помощью пиктограмм, отображение изображений, перемещение изображений, оценка результата перемещения.
3. Проектирование шаблонного интерфейса. Интерфейс игры включает три основные страницы: страница выбора категории, страница задания, страница результатов (рисунок 4.7).





б) Экран выбора ответа



в) Экран завершения

Рисунок 4.7 — Шаблоны страниц игры для пользователей с ТМНР

Дизайн-макет МП создан с использованием онлайн-сервиса Draw.io, результаты сохранены в draw.io файле.

4. Разработка наборов пиктограмм для элементов интерфейса. Каждой странице сопоставлен набор обязательных и опциональных игровых элементов, элементов управления, справочных элементов, каждому элементу сопоставлен predetermined набор возможных визуальных представлений на экране. Для таких элементов интерфейса как кнопки навигации, «Прогресс-бар» пиктограммы выбраны из набора пиктограмм, сохраненных ранее в БД при разработке других МП. Для управления игровым процессом добавлены наборы пиктограмм для отображения категорий и предметов, относящихся к этим категориям, кнопок помощи и кнопок выбора (таблицы 4.1, 4.2).

Таблица 4.1 — Пиктограммы для элементов интерфейса, выбранные из БД

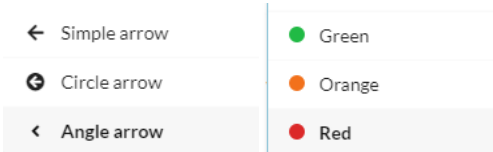
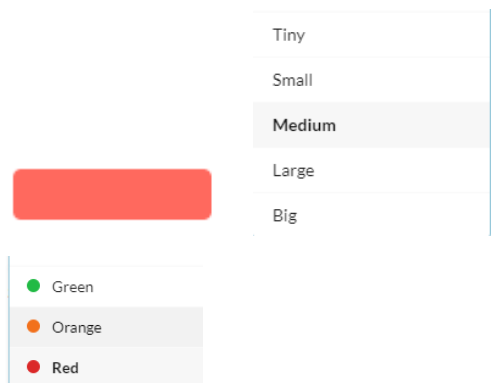
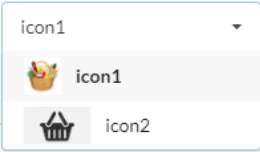
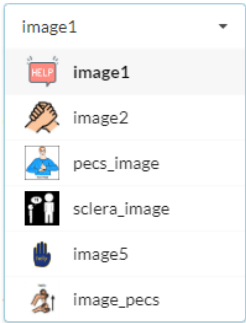
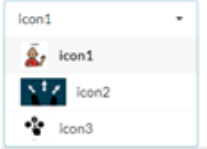
Элемент	Изображения
Кнопки навигации	
Прогресс бар	
Опциональное наличие	

Таблица 4.2 — Пиктограммы, добавленные в БД системы

Элемент	Изображения
Кнопка верной категории	
Кнопка неверной категории	
Кнопка помощи	
Кнопка выбора	

5. Разработка КП для настройки шаблонного интерфейса МП. Для конфигурирования шаблонных страниц экрана МП разработаны соответствующие страницы веб-интерфейса КП. Пример страницы показан на рисунке 4.8 (слева – страница КП; справа – визуализация страницы МП, создаваемой с использованием КП; внизу – фрагмент кода JSX файла с описанием созданной страницы).

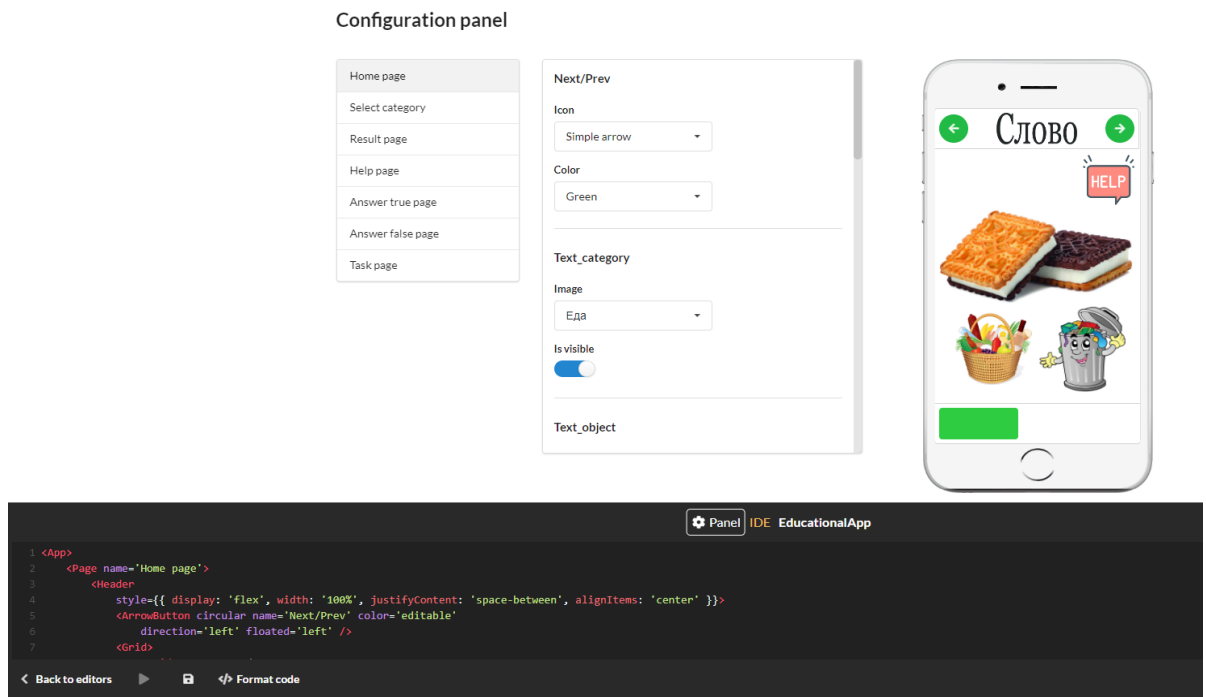


Рисунок 4.8 — КП для главной страницы игры для пользователей с ТМНР

На рисунке 4.9 представлена КП для страницы выбора категории.

Configuration panel

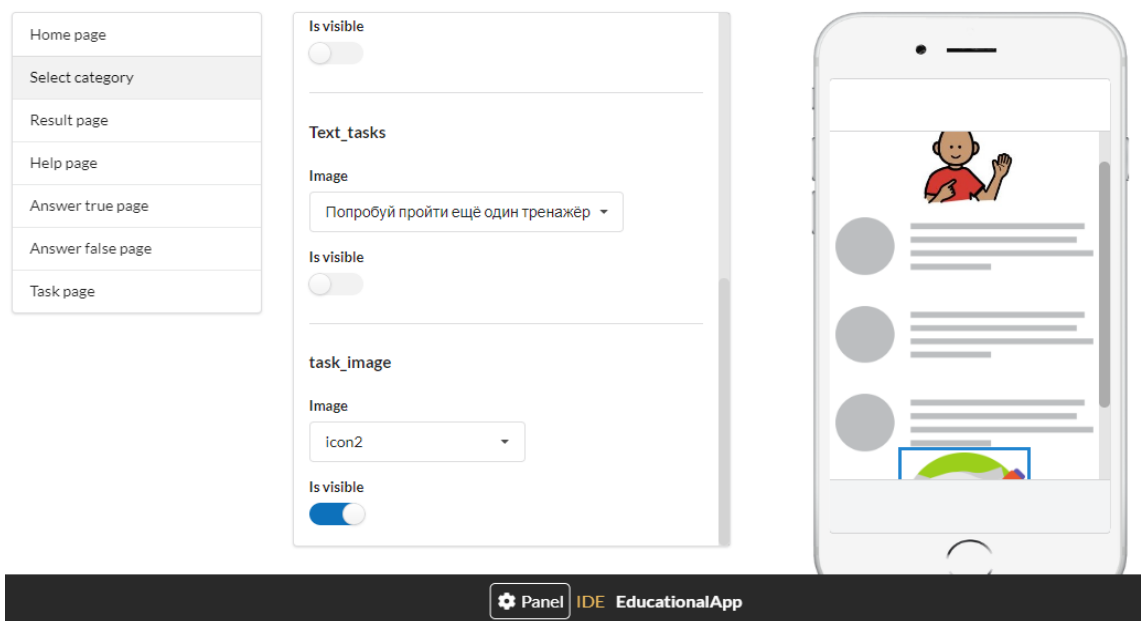


Рисунок 4.9 — КП для страницы выбора категории

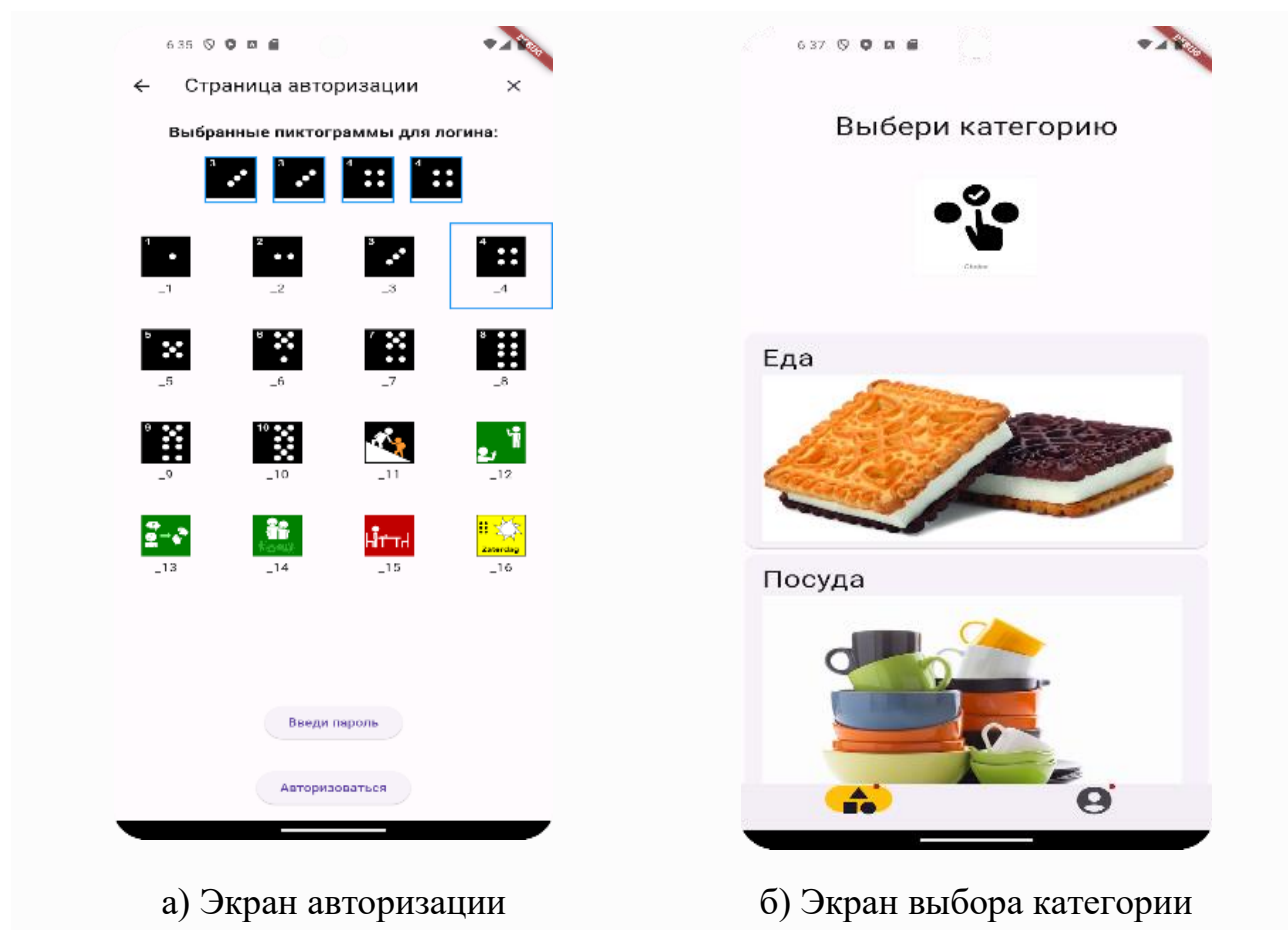
В результате сгенерирован JSX-файл описания шаблонов страниц приложения с обозначенными обязательными и опциональными элементами.

6. Разработка архитектуры игры. Создан проект МП. Из хранилища готовых модулей выбран модуль авторизации с помощью пиктограмм, разработанный ранее для другого МП. Спроектированы feature- и library-модули, реализующие функционал игры.

7. Реализация MVP МП. Реализован проект МП и интегрирован с API CASE-системы.

МП реализовано на языке программирования Dart с использованием фреймворка Flutter. Для разработки МП применена IDE Android Studio.

Варианты экранных форм разработанного МП с возможными элементами интерфейса и пиктограммами, выбранными из БД в процессе адаптации интерфейса, приведены на рисунке 4.10 и 4.11.





в) Экран выбора ответа

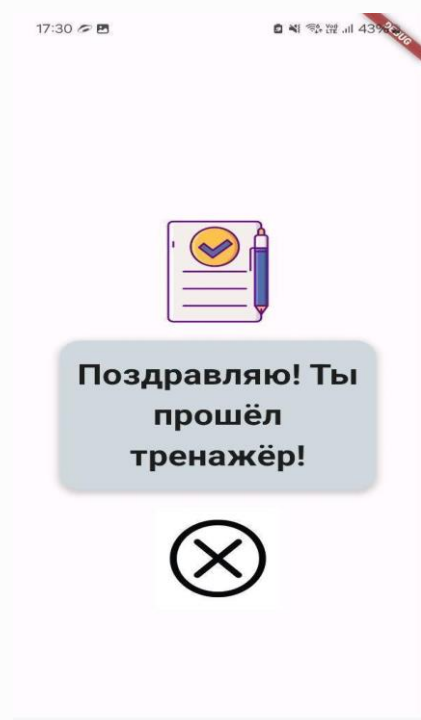


г) Экран выбора ответа

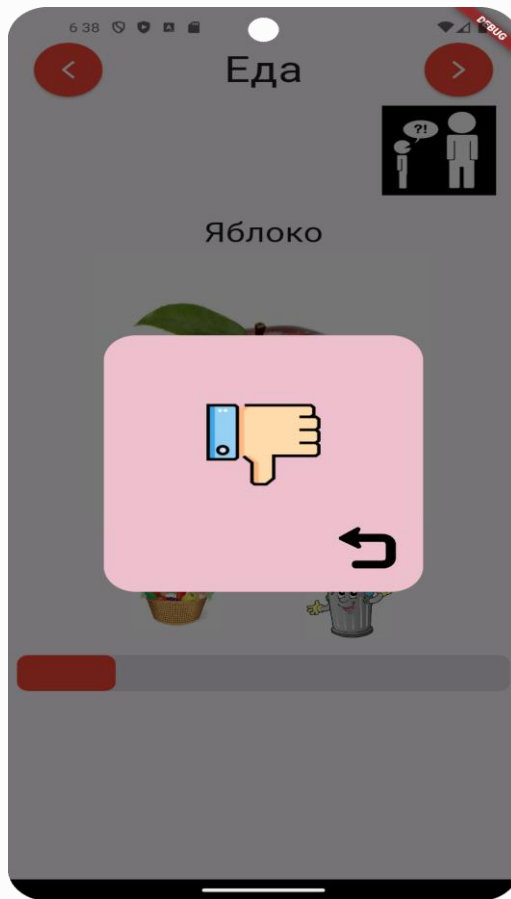
Рис. 4.10 — Экранные формы игры



а) Экран выбора ответа



б) Экран завершения



в) Экран выбранного ответа



г) Экран выбранного ответа

Рис. 4.11 — Экранные формы игры (продолжение)

4.2 Оценка временных затрат на разработку МП в системе

Для оценки временных затрат на разработку МП был проведён анализ трудоёмкости разработки игр (Тф) с применением системы в сравнении с трудоёмкостью разработки такой же игры с применением стандартных средств разработки (Тс).

В таблице 4.3 представлен анализ эффективности системы на этапе разработки МП «Money Game» с применением генерации КП.

Таблица 4.3 - Анализ эффективности системы на этапе разработки МП «Money Game»

Функционал	Трудоёмкость разработки, час.		
	Т _ф	Комментарий	Т _с
Разработка архитектуры	1	Использован шаблон многомодульной архитектуры	8
Разработка экрана авторизации	1	Выбран готовый модуль из хранилища модулей	8
Разработка экрана выбора дня недели	6	Для некоторых элементов управления выбраны готовые пиктограммы из БД	8
Разработка экрана меню	6		8
Разработка экрана окончания работы с меню	6		8
Разработка КП	5	КП позволяет конфигурировать МП под различных пользователей	0
Тестирование	4		4
Итого	29		44

Оценка временных затрат на разработку МП Money Game представлена на рисунке 4.12.

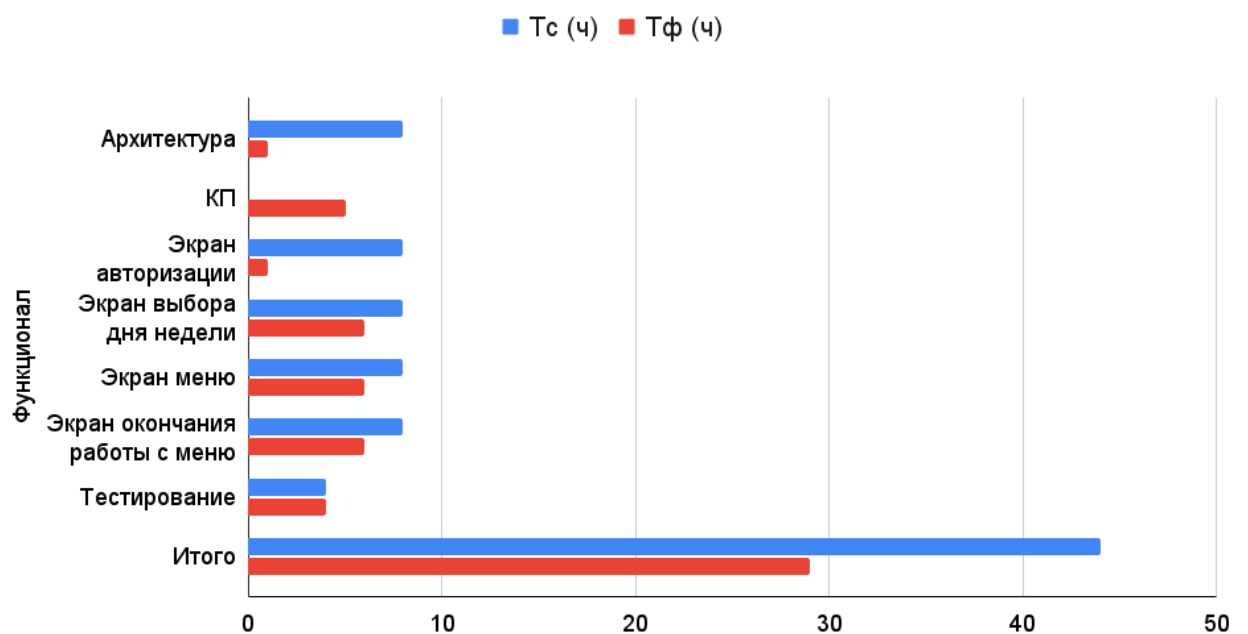


Рисунок 4.12 — Оценка временных затрат на разработку МП Money Game

Использование разработанной системы позволило сократить время разработки интерфейса МП примерно в 1.5 раза, а самого МП почти в 2 раза.

В таблице 4.4 представлен анализ эффективности системы на этапе разработки МП «AIT Test» с применением генерации КП.

Таблица 4.4 — Анализ временных затрат на разработку МП «AIT Test»

Функционал	Трудоёмкость разработки, час.		
	Т _ф	Комментарий	Т _с
Разработка архитектуры	1	Использован шаблон многомодульной архитектуры	8
Разработка экрана авторизации	1	Выбран готовый модуль из хранилища модулей	8
Разработка экрана обучения	4	Для некоторых элементов управления выбраны готовые пиктограммы из БД	6
Разработка экрана тестирования	4		6
Разработка экрана предварительного результата	2		4
Разработка экрана окончательного результата	2		4
Разработка экрана завершения	2		4
Разработка КП	5	КП позволяет конфигурировать МП под различных пользователей	0
Тестирование	4		4
Итого	25		44

Оценка временных затрат на разработку МП “AIT Test” представлена на рисунке 4.13.

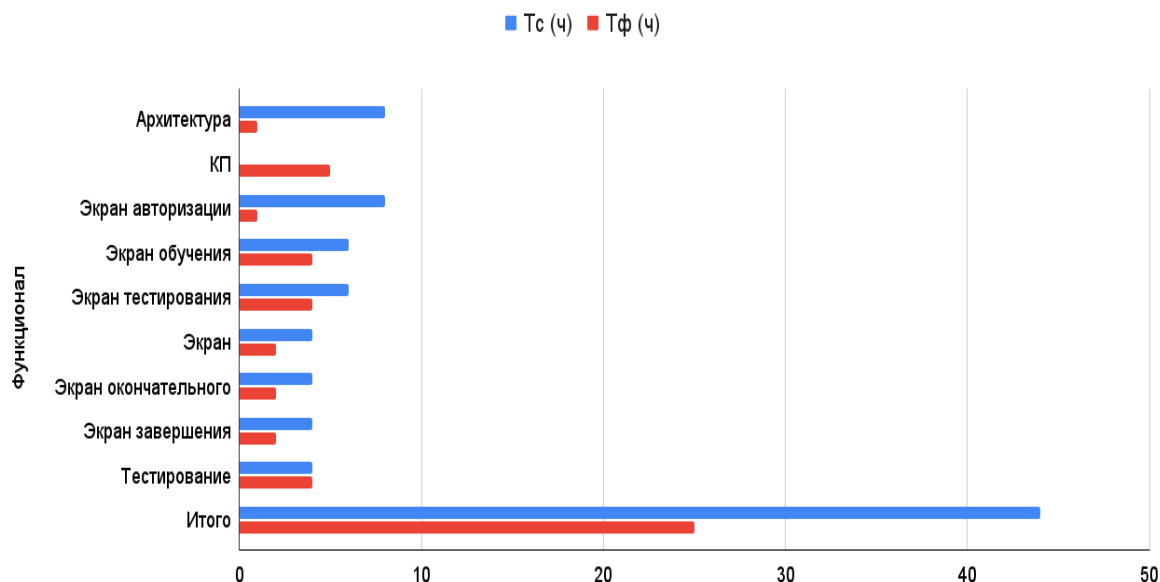


Рисунок 4.13 — Оценка временных затрат на разработку МП “AIT Test”

Использование разработанной системы позволило сократить время разработки интерфейса МП более чем в 2 раза, а самого МП почти в 2 раза.

В таблице 4.5 представлен анализ эффективности системы на этапе разработки МП для людей с ТМНР с применением генерации КП.

Таблица 4.5 — Оценка временных затрат на разработку МП для людей с ТМНР

Функционал		Трудоёмкость разработки, час.		
		Т _ф	Комментарий	Т _с
Разработка архитектуры		1	Использован шаблон многомодульной архитектуры	8
Разработка экранов интерфейса	Разработка экрана авторизации	1	Выбран готовый модуль из хранилища модулей	8
	Разработка экрана выбора категории	4	Для элементов управления выбраны пиктограммы из БД	5
	Разработка экрана задания	6		8
	Разработка экрана проверки ответа	2		3

	Разработка экрана окончания	2		4
Разработка КП		5		0
Тестирование		4		4
Итого		25		40

Оценка временных затрат на разработку МП для пользователей с ТМНР представлена на рисунке 4.14.

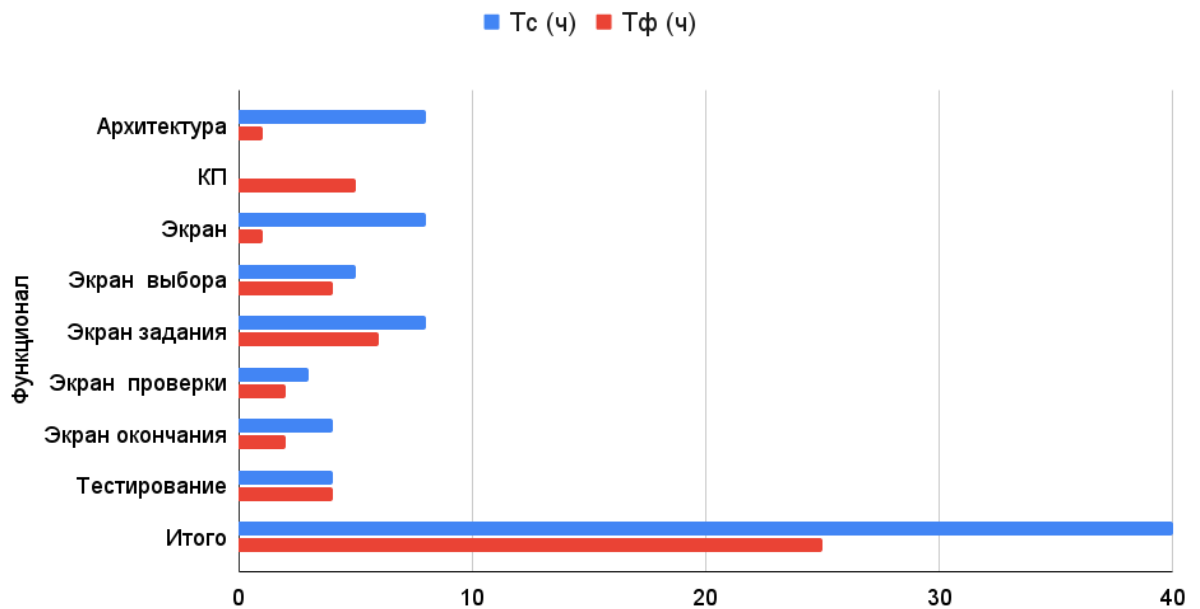


Рисунок 4.14 — Оценка временных затрат на разработку МП для пользователей с ТМНР

Использование разработанной системы позволило сократить время разработки интерфейса МП более чем в 2 раза, а самого МП почти в 2 раза.

Также был проведен анализ временных затрат на разработку всех МП. Оценка временных затрат общего времени разработки трех МП представлена на рисунке 4.15.

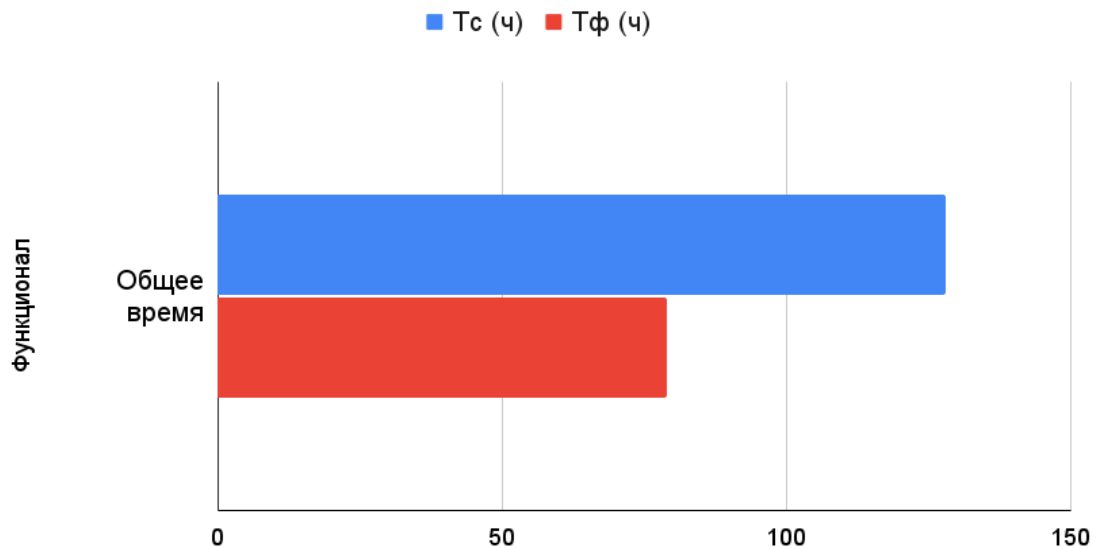


Рисунок 4.15 — Оценка временных затрат общего времени разработки трех МП

Для всех МП добавилось время на разработку КП, но за счет сокращения времени на все остальные процедуры разработки общее время сократилось. При этом применение КП позволило существенно сокращать время на конфигурирование одного и того же МП под различных пользователей с разными возможностями и ограничениями. Кроме того, разработка каждого нового МП приводила к пополнению хранилища модулей для повторного использования, поэтому время на разработку каждого последующего МП сокращалось. Суммарное время разработки МП с применением системы сократилось примерно в два раза.

4.3 Оценка доступности МП

Для оценки доступности разработанных МП для людей с ИН был использован метод внешнего наблюдения за пользователями МП и оценки их способности самостоятельного управления МП. Сотрудники, работающие с людьми с ИН, наблюдали за действиями своих подопечных и делали выводы о том, насколько успешно пользователи справлялись с управлением МП.

После разработки версий МП с адаптируемым интерфейсом было проведено повторное тестирование с сотрудниками и пользователями с ИН в

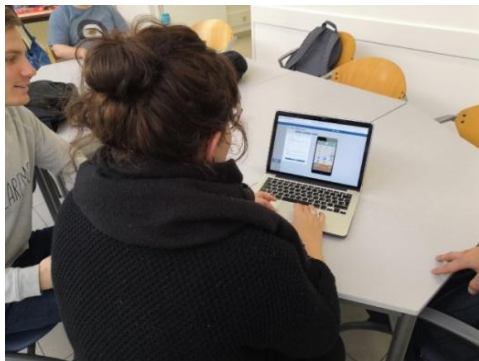
Волгоградском психоневрологическом интернате и медицинском центре MRI Oosterlo, Бельгия. Сотрудники настраивали интерфейсы МП для каждого пользователя с ИН, наблюдали за действиями пользователей МП и делали выводы о том, насколько успешно пользователи управляли МП. Процесс тестирования показан на рисунке 4.16.



а) Настройка интерфейсов МП сотрудниками MRI Oosterlo



б) Настройка интерфейсов МП сотрудниками MRI Oosterlo



в) Настройка интерфейсов МП сотрудниками MRI Oosterlo



г) Тестирование МП с клиентами с ТМНР в Волгоградском психоневрологическом интернате

Рисунок 4.16 — Тестирование с пользователями CASE-системы и пользователями

Сотрудники отметили, что пациенты с большим интересом участвовали в тестировании, практически все они смогли управлять МП без помощи (или с минимальной помощью) сотрудников. Успешная работа пользователей с ИН с МП показала, что настройка интерфейса МП для каждого пользователя

сотрудниками, которые хорошо знают своих подопечных, обеспечила доступность МП для пользователей с разными возможностями и ограничениями. В целом, результаты тестирования показали, что применение веб-системы позволяет разрабатывать МП, помогающие людям с ИН развивать различные навыки повседневной жизни и социализироваться в современном обществе. Таким образом, адаптация интерфейсов МП под каждого пользователя обеспечила доступность МП пользователям с разными возможностями и ограничениями.

4.4 Выводы

В четвертой главе проведена апробация системы на разработке трех МП, оценены временные затраты, доступность МП и получены следующие результаты:

- система помогла при разработке МП "AIT TEST", "Money Game", МП для людей с ТМНР за счет использований КП и модуляризации;
- суммарное время разработки МП с применением системы сократилось примерно в два раза, а разработка каждого нового МП приводила к пополнению хранилища модулей для повторного использования, поэтому время на разработку каждого последующего МП сокращалось;
- адаптация интерфейсов МП под каждого пользователя обеспечила доступность МП пользователям с разными возможностями и ограничениями, а сотрудники специализированных учреждений отметили, что пациенты с большим интересом участвовали в тестировании, практически все они смогли управлять МП без помощи (или с минимальной помощью) сотрудников.

Заключение

Выполненные исследования и практическая работа позволили получить следующие выводы и результаты:

1. диссертация представляет научно-квалификационную работу, в которой содержится решение задачи по повышению эффективности процесса проектирования МП для пользователей с ИН и снижению трудоемкости разработки МП за счет автоматизации проектных и технологических процедур;
2. анализ современных подходов к автоматизации проектирования МП для людей с ИН показал, что большинство из них базируются на принципах универсального дизайна и адаптируемого интерфейса, а оценка доступности основана либо на априорном экспертном оценивании на соответствие принципам доступности, либо на внешнем наблюдении за действиями пользователя с уже адаптированным интерфейсом, что не обеспечивает комплексного решения проблемы. В связи с этим для обеспечения доступности МП для широкого круга пользователей с ИН, возможности которых сильно различаются, требуется разработка новой модели ЖЦ, учитывающей особенности пользователей, а также моделей, методов и технологий проектирования, поддерживающих не только параметрическую, но и структурную адаптацию интерфейса, что необходимо предусматривать на этапе проектирования и реализовывать в специализированных CASE-средствах сквозной поддержки ЖЦ МП для данной категории пользователей;
3. выявлены связи между характеристиками пользователей с ИН и требованиями к параметрической и структурной адаптации интерфейса МП, что дало возможность разработать модель ЖЦ МП для пользователей с ИН, отражающую специфику такой категории МП и предложить способ формализации проектных и технологических процедур этапов ЖЦ МП для пользователей с ИН, обеспечивающий сквозную поддержку этапов

проектирования, реализации, адаптации и эксплуатации;

4. предложен шаблон проектирования МП, позволяющий снизить трудоемкость проектирования МП и повысить качество проектных решений;
5. предложен метод проектирования интерфейсов МП, позволяющий адаптировать шаблонный интерфейс под конечного пользователя на этапе эксплуатации без изменения программного кода;
6. предложенный метод автоматизированного проектирования МП реализован в специализированной CASE-системе, обеспечивающей сквозную поддержку этапов проектирования, реализации и эксплуатации ЖЦ МП;
7. проведена апробация CASE-системы на разработке 4-х МП для пользователей с ИН (квалификация разработчиков – студенты 2-4 курсов бакалавриата). Разработка МП с применением системы позволила снизить трудоёмкость процесса проектирования МП в среднем в 2 раза;
8. проведено тестирование CASE-системы в режиме настройки интерфейсов разработанных МП под конечных пользователей с В. тестировании участвовали 4 сотрудника Волгоградского психоневрологического интерната (<https://442fz.volganet.ru/025001/>) и 3 сотрудника медицинского центра MPI Oosterlo, Бельгия (<https://www.mpi-oosterlo.be/>), специализирующиеся на работе с людьми с ИН. Тестирование показало, что все сотрудники успешно смогли настроить интерфейсы для своих подопечных с учетом их возможностей и ограничений;
9. проведено тестирование МП, разработанных с применением CASE-системы, с 6-ю пользователями с ИН. Тестирование показало, что адаптация интерфейсов МП под каждого пользователя с учетом его возможностей и ограничений обеспечила доступность МП для всех пользователей, участвующих в тестировании.

В целом, разработанные модели, методы, алгоритмы и система позволяют усовершенствовать процесс разработки адаптивных МП на этапах

проектирования и реализации, существенно снизить трудоемкость разработки таких МП и повысить их качество за счет применения апробированных проектных и технологических решений. Таким образом, в ходе выполнения диссертационной работы достигнута цель исследования и решены поставленные задачи.

Применение системы для разработки МП для пользователей с ИН должно позволить существенно сократить временные затраты на разработку за счет использования унифицированной многомодульной архитектуры и включения в проект МП готовых модулей, а также генерации КП для адаптации интерфейса МП. Возможность применения шаблонов архитектур и готовых решений расширит круг пользователей системы и сделает ее доступной разработчикам, профессиональная деятельность которых связана с поддержкой людей с ограниченными интеллектуальными возможностями. Привлечение разработчиков МП к разработке приложений для людей с ИН с применением системы позволит пополнять хранилище модулей, доступных для подключения в новые проекты, и расширять, таким образом, возможности системы.

В качестве перспектив дальнейшей работы можно выделить масштабирование системы и массовое внедрение в медицинские центры и другие соответствующие организации.

Список литературы

1. World Health Organization [Электронный ресурс] – Режим доступа: <https://www.who.int>.
2. Инвалидность в России и в мире в цифрах [Электронный ресурс] – Режим доступа: <https://tiflocentre.ru/stati/statistika-po-invalidnosti.php>.
3. Федеральный закон "О социальной защите инвалидов в Российской Федерации" от 24.11.1995 N 181-ФЗ (последняя редакция).
4. Sheehan R., Hassiotis A. Digital mental health and intellectual disabilities: state of the evidence and future directions // Evidence Based Mental Health. – 2017. – Vol. 20, No. 4. – P. 107–111. – DOI: 10.1136/eb-2017-102759.
5. Social Inclusion and Adults with Intellectual Disability: The Role of Mobile Technology [Электронный ресурс]. Режим доступа: <https://www.unsw.edu.au/content/dam/pdfs/ada/education/research-reports/2022-07-project-briefs/2022-07-15-ada-social-inclusion-and-adults-with-intellectual-disability.pdf>.
6. Alanazi A. Smartphone apps for transportation by people with intellectual disabilities: are they really helpful in improving their mobility? // Disability and Rehabilitation: Assistive Technology. – 2022. – Vol. 17, No. 1. – P. 1–7. – DOI: 10.1080/17483107.2020.1820085.
7. Korczak P., Zwierzchowska A. Using mobile applications in the process of enhancing and restoring abilities in individuals with intellectual disability and other disabilities – a literature review // Advances in Rehabilitation. – 2020. – Vol. 34, No. 4. – P. 36–41. – DOI: 10.5114/areh.2020.100774.
8. Корнева О. А. Использование мобильных приложений для социальной адаптации людей с ограниченными возможностями // Социология и право. – 2022. – № 4.
9. Макуха Л. В., Котов С. А., Казакова У. В. Мобильное приложение для людей с ограниченными возможностями // Международный журнал гуманитарных и естественных наук. – 2022. – № 6-1.

10. Mayordomo-Martínez D., Carrillo-de-Gea J. M., García-Mateos G., García-Berná J. A., Fernández-Alemán J. L., Rosero-López S., Parada-Sarabia S., García-Hernández M. Sustainable Accessibility: A Mobile App for Helping People with Disabilities to Search Accessible Shops // *International Journal of Environmental Research and Public Health*. – 2019. – Vol. 16, No. 4. – P. 620. – DOI: 10.3390/ijerph16040620.
11. Hardiyanti M., Pratomo D. N., Krisnandaru R. A. D., Riona V. Accessibility evaluation of interactive learning mobile applications for individuals with intellectual disabilities // *International Journal of Information and Education Technology*. – 2024. – Vol. 14, No. 3. – P. 426–434.
12. Gullà F., Ceccacci S., Germani M., Cavalieri L. Design adaptable and adaptive user interfaces: a method to manage the information // *Biosystems and Biorobotics*. – 2014. – Vol. 11. – P. 47–58. – DOI: 10.1007/978-3-319-18374-9_5.
13. Часовских Ю., Бакаев М. Проектирование веб-интерфейсов для пользователя на основе конфигурируемых компонентов // *Информационное общество: образование, наука, культура и технологии будущего*. – 2018. – № 2. – С. 75–83. – DOI: 10.17586/2587-8557-2018-2-75-83.
14. Международная классификация болезней 10-го пересмотра (МКБ-10) [Электронный ресурс] – Режим доступа: <https://mkb-10.com/>.
15. Jobe W. Native Apps Vs. Mobile Web Apps // *International Journal of Interactive Mobile Technologies (iJIM)*. – 2013. – Vol. 7, No. 4. – P. 27–32. – DOI: 10.3991/ijim.v7i4.3226.
16. Нгуен Б. Х., Нгуен Д. Ч., Нгуен Н. Х. и др. Виды мобильных приложений, что способствует улучшению пользовательского опыта // *Синергия Наук*. – 2017. – № 7. – С. 246–253.
17. Pani S., Mishra J. A novel approach for mobile native app development using ontological design // *International Journal of Software Engineering and Its Applications*. – 2016. – Vol. 10, No. 9. – P. 105–124. – DOI: 10.14257/ijseia.2016.10.9.10.

18. Bal J. S. Development of native mobile application // *Journal of Critical Reviews*. – 2021. – Vol. 8, Issue 3.
19. Юнкникель К. М. Инструменты кроссплатформенной разработки мобильных приложений // *Студенческая наука и XXI век*. – 2023. – Т. 20, № 1-1(23). – С. 113–115.
20. Nagesh A., Caicedo C. Cross-platform mobile application development [Электронный ресурс] – Режим доступа: https://www.researchgate.net/publication/263416908_Cross-Platform_Mobile_Application_Development.
21. Amatya S., Kurti A. Cross-platform mobile development: challenges and opportunities // *ICT Innovations 2013* / ed. by V. Trajkovik, M. Anastas. – Cham: Springer, 2014. – Vol. 231 (Advances in Intelligent Systems and Computing). – P. 219-229. – DOI: 10.1007/978-3-319-01466-1_21.
22. Latif M., Lakhrissi Y., Nfaoui E.H., Es-Sbai N. Cross platform approach for mobile application development: a survey // *2016 International Conference on Information Technology for Organizations Development (IT4OD)*. – 2016. – P. 1-5. – DOI: 10.1109/IT4OD.2016.7479278.
23. Васильев Н.П. Гибридные технологии разработки приложений для мобильных платформ // *Информационные системы и технологии: теория и практика: сб. науч. тр. науч.-техн. конф. ин-та леса и природопользования, Санкт-Петербург, 1 февр. 2017 г. / С.-Петерб. гос. лесотехн. ун-т им. С.М. Кирова*. – СПб., 2017. – С. 12-21.
24. Vilcek T., Jakopc T. Comparative analysis of tools for development of native and hybrid mobile applications // *40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. – 2017. – P. 1516-1521. – DOI: 10.23919/MIPRO.2017.7973662.
25. Biørn-Hansen A., Majchrzak T. A., Grønli T.-M. Progressive web apps for the unified development of mobile applications // *Web Information Systems and Technologies*. – P. 64-86. DOI: 10.1007/978-3-319-93527-0_4.

26. Хузяханов Е. И., Шарко С. С., Поляков П. М. Прогрессивные веб-приложения (PWA) как будущее веб-разработки // Современные научные исследования и инновации. — 2024. — № 4(156).
27. Progressive Web Apps (PWA) // International Journal of Emerging Technologies and Innovative Research. — 2021. — Vol. 8, Issue 7. — P. e696–e706. — ISSN: 2349-5162.
28. Correia F., Ribeiro Ó., Silva J. C. Progressive Web Apps Development: Study of Caching Mechanisms // 2021 21st International Conference on Computational Science and Its Applications (ICCSA): proceedings, Cagliari, Italy, 2021. — P. 181–187. — DOI: 10.1109/ICCSA54496.2021.00033.
29. Fauzan R., Krisnahati I., Nurwibowo B., Wibowo D. A systematic literature review on progressive web application practice and challenges // IPTEK The Journal for Technology and Science. — 2022. — Vol. 33. — P. 43 — DOI: 10.12962/j20882033.v33i1.13904.
30. Жидков И. В., Зубарев И. В., Хабибуллин И. В. Выбор рациональной модели разработки защищенного программного обеспечения // Вопросы кибербезопасности. — 2021. — № 5(45). — С. 21–29.
31. Холмогоров В. В., Баранюк В. В. Выбор стандартов в соответствии с этапами жизненного цикла информационных систем // International Journal of Open Information Technologies. — 2022. — № 3.
32. Pargaonkar Shravan. A comprehensive research analysis of software development life cycle (SDLC) Agile & Waterfall model advantages, disadvantages, and application suitability in software quality engineering // International Journal of Scientific and Research Publications. — 2023. — Vol. 13, Issue 8. — P. 120–124. — DOI: 10.29322/IJSRP.13.08.2023.p14015.
33. Jharko E. Life cycle and quality assurance of software for systems of critical information infrastructure facilities // 2021 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM): proceedings, Sochi, Russia, 2021. — P. 440–444. — DOI: 10.1109/ICIEAM51226.2021.9446409.

34. Kyeremeh Kwadwo. Overview of system development life cycle models // SSRN Electronic Journal. — 2019. — DOI: 10.2139/ssrn.3448536.
35. Gurung Gagan, Shah Rahul, Jaiswal Dhiraj. Software development life cycle models – A comparative study // International Journal of Scientific Research in Computer Science, Engineering and Information Technology. — 2020. — Vol. 6, Issue 4. — P. 30–37. — DOI: 10.32628/CSEIT206410.
36. Shokhista E., Kruglov A. Software Development Life Cycle early phases and quality metrics: A systematic literature review // Journal of Physics: Conference Series. - 2020. - Vol. 1694. - Art. 012007. - DOI: 10.1088/1742-6596/1694/1/012007.
37. Kyeremeh K. Overview of System Development Life Cycle Models // SSRN Electronic Journal. - 2019. — DOI: 10.2139/ssrn.3448536.
38. Natara R., Jagatheesan S. Lack of SDLC models and frameworks in mobile application development: A systematic literature review and study. - 2021.
39. Vagrani A., Singh N., Vigneswara P. Decline in mobile application life cycle // Procedia Computer Science. - 2017. - Vol. 122. - P. 1080-1087. - DOI: 10.1016/j.procs.2017.11.460.
40. Kaur A., Kaur K. Suitability of existing software development life cycle (SDLC) in context of mobile application development life cycle (MADLC) // International Journal of Computer Applications. - 2015. - Vol. 123, No. 12. - P. 1-5. - DOI: 10.5120/20441-2785.
41. Jabangwe R., Edison H., Duc A. N. Software engineering process models for mobile app development: A systematic literature review // Journal of Systems and Software. — 2018. — Vol. 145. — P. 98–111. — DOI: 10.1016/j.jss.2018.08.028.
42. Inukollu V. N., Keshamoni D., Kang T., Inukollu M. Factors influencing quality of mobile apps: Role of mobile app development life cycle // International Journal of Software Engineering & Applications (IJSEA). — 2014. — Vol. 5, No. 5. — P. 15–34. — DOI: 10.5121/ijsea.2014.5502.
43. Vagrani A., Singh N., Ilavarasan P. V. Decline in mobile application life cycle // Procedia Computer Science. — 2017. — Vol. 122. — P. 957–964. — DOI: 10.1016/j.procs.2017.11.460.

44. Жизненный цикл проекта по созданию мобильного приложения [Электронный ресурс] – Режим доступа: <https://projecto.pro/blog/theory/zhiznennyj-czikl-proekta-po-sozdaniyu-mobilnogo-prilozheniya/>.

45. Жизненный цикл разработки мобильного ПО [Электронный ресурс] – Режим доступа: <https://learn.microsoft.com/ru-ru/previous-versions/xamarin/cross-platform/get-started/introduction-to-mobile-sdlc>.

46. Артюшкин О. В., Хрусталева В. И. Качественные характеристики сложности разработки программных систем // Вестник Хакасского государственного университета им. Н. Ф. Катанова. — 2017. — Т. 9, № 20 (4). — С. 5–8.

47. Хорошевский В. Ф. Проектирование систем программного обеспечения под управлением онтологий: модели, методы, реализации // Онтология проектирования. — 2019. — № 4 (34).

48. Kapila G., Dias A. Evolvment of Computer Aided Software Engineering (CASE) Tools: A User Experience // International Journal of Computer Science and Software Engineering (IJCSE). — 2017. — Vol. 6, No. 3. — P. 55–60.

49. Вичугова А. А. Автоматизация процесса разработки программного обеспечения: методы и средства // Прикладная информатика. — 2016. — № 3 (63). — С. 63–75.

50. Hossain Mohammad. Software Development Life Cycle (SDLC) Methodologies for Information Systems Project Management // International Journal For Multidisciplinary Research. — 2023. — Vol. 5, No. 5. – DOI: 10.36948/ijfmr.2023.v05i05.6223.

51. Gurung G., Shah R., Jaiswal D. Software Development Life Cycle Models – A Comparative Study // International Journal of Scientific Research in Computer Science, Engineering and Information Technology. — 2020. — Vol. 6, No. 4. — P. 30–37. — DOI: 10.32628/CSEIT206410.

52. Куликова Е. В. Инструментальные средства информационной поддержки жизненного цикла программного обеспечения // Современные материалы, техника и технологии. — 2017. — № 3 (11).
53. Burnashev R. A., Gubajdullin A. V., Enikeev A. I. Specialized case tools for the development of expert systems // Trends and Advances in Information Systems and Technologies: Proceedings of the WorldCIST'18 (eds. Á. Rocha et al.). — 2018. — Vol. 745. — P. 599-605.
54. Shafie S., Wautelet Y., Friis S. C., Lis L., Harlou U., Hvam L. Evaluating the benefits of a computer-aided software engineering tool to develop and document product configuration systems // Computers in Industry. — 2021. — Vol. 128. — Art. 103432. — DOI: 10.1016/j.compind.2021.103432.
55. Enikeev A. I., Georgiev V. O., Mohammed M. B. A. Specialized case tools for the development of the accounting applications // A.I. Journal of Fundamental and Applied Sciences: Special Issue. — 2017. — Vol. 9, No. 25.
56. Imam A., Alnsour A., Alhroob A. The Definition of Intelligent Computer Aided Software Engineering (I-CASE) Tools // Journal of Information Engineering and Applications. — 2015. — Vol. 5, No. 6. — P. 47-56.
57. Sisala S., Othman S. Developing a Mobile Device Management (MDM) Security Metamodel for Bring Your Own Devices (BYOD) in Hospitals // International Journal of Innovative Computing. — 2020. — Vol. 10, No. 2. — DOI: 10.11113/ijic.v10n2.273.
58. Yamin M., Katt B. Mobile device management (MDM) technologies, issues and challenges // Proceedings of the 3rd International Conference on Cryptography, Security and Privacy (ICCSP '19). — 2019. — P. 143-147. — DOI: 10.1145/3309074.3309103.
59. Rhee K., Jeon W., Won D. Security Requirements of a Mobile Device Management System // International Journal of Security and Its Applications. — 2012. — Vol. 6, No. 2. — P. 353-358.
60. Mobile application management (MAM) [Электронный ресурс] — Режим доступа:

<https://www.techtarget.com/searchmobilecomputing/definition/mobile-application-management-MAM>.

61. What is Microsoft Intune app management? [Электронный ресурс] – Режим доступа: <https://learn.microsoft.com/en-us/mem/intune/apps/app-management>.

62. VVisariya J., Dube M. Enterprise Mobile Application Management Platform - A Hybrid Approach // International Journal of Engineering Research and Technology. — 2015. — Vol. 4, Issue 05. — DOI: 10.17577/IJERTV4IS050351.

63. Mobile Device Management Enterprise Mobility Management Управление мобильными устройствами [Электронный ресурс] :офиц. сайт. – Режим доступа: [http://www.tadviser.ru/index.php/Mobile_Device_Management_\(MDM\)](http://www.tadviser.ru/index.php/Mobile_Device_Management_(MDM)).

64. Hayes D., Cappa F., Le-Khac N. An Effective Approach to Mobile Device Management: Security and Privacy Issues Associated with Mobile Applications // Digital Business. — 2020. — Vol. 1. — Art. 100001. — DOI: 10.1016/j.digbus.2020.100001.

65. IBM Mobile First [Электронный ресурс] – Режим доступа: <https://www.ibm.com/docs/en/cloud-private/3.2.x?topic=paks-mobilefirst-platform-foundation>.

66. Jamf pricing [Электронный ресурс] – Режим доступа: <https://www.jamf.com/pricing/>.

67. OKTA [Электронный ресурс] – Режим доступа: <https://www.okta.com/>.

68. Mobilock Pro [Электронный ресурс] – Режим доступа: <https://mobilockpro.com/>.

69. ManageEngine [Электронный ресурс] – Режим доступа: <https://www.manageengine.com/mobile-device-management/>.

70. Appery.io [Электронный ресурс] – Режим доступа: <https://appery.io/>

71. TheAppBuilder [Электронный ресурс] – Режим доступа: <https://m.theappbuilder.com/Account/Login>.

72. iBuildApp [Электронный ресурс] – Режим доступа: <https://russia.ibuildapp.com/>.
73. AppsBuilder [Электронный ресурс] – Режим доступа: <https://appsbuilder.io/>.
74. Shabalina O., Guriev V., Kosyakov S., Dmitriev N., Davtian A. MADM System for the Development of Adaptable Mobile Applications for People with Intellectual Disabilities // 2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA): Proceedings, Piraeus, Greece, July 15-17, 2020. — P. 1-8. — DOI: 10.1109/IISA50023.2020.9284376.
75. WHO. The Global Burden of Disease: 2004 Update. [Электронный ресурс] — Режим доступа: http://www.who.int/healthinfo/global_burden_disease/2004_report_update/en/.
76. Tizard Centre, University of Kent at Canterbury. Intellectual Disability in Europe: Working Papers. 2003.
77. Maulik P. K., Mascarenhas M. N., Mathers C. D., Dua T., Saxena S. Prevalence of Intellectual Disability: A Meta-Analysis of Population-Based Studies // Research in Developmental Disabilities. — 2011. — Vol. 32, No 2. — P. 419-436. — DOI: 10.1016/j.ridd.2010.12.018.
78. Definition of Intellectual Disability [Электронный ресурс] – Режим доступа: <https://www.aaid.org/intellectual-disability/definition>.
79. Shree A., Shukla P. Intellectual Disability: Definition, Classification, Causes and Characteristics // Learning Community-An International Journal of Educational and Social Development. - 2016. - Vol. 7, No 1. - P. 9-20. - DOI: 10.5958/2231-458X.2016.00002.6.
80. American Psychiatric Association. Diagnostic and Statistical Manual of Mental Disorders (DSM-5). - 5th ed. - Washington, DC: American Psychiatric Publishing, 2013.
81. World Health Organization. International Statistical Classification of Diseases and Related Health Problems, 10th Revision, 2nd Edition. Geneva: World Health Organization. – 2004.

82. Башаев С. В., Горелик В. В. Проблемы социальной адаптации лиц с ограниченными возможностями здоровья в современном обществе // Международный научно-исследовательский журнал. - 2016. - № 4-7 (46).
83. Похвощев В. А., Колесникова О. А., Фирсова Ю. А. Методологические основы профессиональной реабилитации и содействия занятости людей с ограниченными возможностями здоровья // МИР (Модернизация. Инновации. Развитие). – 2017. – № 2 (30).
84. Попов А. Н. Роль информационных технологий как средств социальной поддержки детей-инвалидов // Вестник ТГУ. – 2013. – № 11 (127).
85. Истархова В. Н., Корнеева Р. В. Информационные технологии для лиц с ОВЗ // Базовые тренды социально-экономического развития: вопросы оценки: материалы регион. науч.-практ. конф., Калуга, 2021 г. - Калуга. 2021. - С. 284-293.
86. Маллаев Д. М., Бажукова О. А., Болдинова О. Г. Информационные и коммуникационные технологии в специальном и инклюзивном образовании // Известия Дагестанского государственного педагогического университета. Психолого-педагогические науки. – 2021. – Т. 15. – № 3. – С. 42-46.
87. Marchlewska A., Goetzen P. The Role of Information Technology for People with Intellectual Disabilities // Journal of Applied Computer Science Methods. - 2014. - Vol. 6, No 1. DOI:10.2478/jacsm-2014-0006.
88. Almufareh M. F., Tehsin S., Humayun M. [и др.] Intellectual Disability and Technology: An Artificial Intelligence Perspective and Framework // Journal of Disability Research (JDR). - 2023. - Vol. 2, No 4. - P. 58-70. - DOI: 10.57197/JDR-2023-0055.
89. Boot F., Julia S., Hung J., Chen R.. Editorial: Intellectual Disability and Assistive Technology // Frontiers in Public Health. – 2019.– Vol. 7.
90. Sheehan R., Hassiotis A. Digital mental health and intellectual disabilities: state of the evidence and future directions // Evidence-Based Mental Health. - 2017. - Vol. 20, No 4. - P. 107-111. - DOI: 10.1136/eb-2017-102759.

91. Zaina L. A. M., Fortes R. P. M., Casadei V. [и др.]. Preventing Accessibility Barriers: Guidelines for Using User Interface Design Patterns in Mobile Applications // Journal of Systems and Software. – 2022. – Vol. 186.
92. da Silva L. F., Parreira Junior P. A., Freire A. P. Mobile User Interaction Design Patterns: A Systematic Mapping Study // Information. – 2022.
93. Bennett R., Vijaygopal R. Adoption of New Transportation Assistive Technologies by People with Mild Intellectual Disabilities // Disability and Rehabilitation: Assistive Technology. – 2023. DOI:10.1080/17483107.2023.2286522.
94. Zhao Y., Gong L., Yang W., Zhou Y. How Accessibility Affects Other Quality Attributes of Software? A Case Study of GitHub // Science of Computer Programming. - 2024. - Vol. 231. - Art. 103027. - ISSN 0167-6423. - DOI: 10.1016/j.scico.2023.103027.
95. Ara J., Sik-Lanyi C., Kelemen A. Accessibility Engineering in Web Evaluation Process: A Systematic Literature Review // Universal Access in the Information Society. - 2024. - Vol. 23. - P. 653-686. - DOI: 10.1007/s10209-023-00967-2.
96. Borblik Y., Shabalina O., Kultsova M., Pidoprigora A., Romanenko R. Assistive Technology Software for People with Intellectual or Development Disabilities: Design of User Interfaces for Mobile Applications // 2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA): Proceedings, Corfu, Greece, 6-8 July 2015. - Piscataway: IEEE, 2015. - DOI: 10.1109/IISA.2015.7387976.
97. Ballantyne M., Jha A., Jacobsen A., Hawker J. S., El-Glaly Y. N. Study of Accessibility Guidelines of Mobile Applications // Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia (MUM 2018), Cairo, Egypt, November 25-28, 2018. - New York: ACM, 2018. - P. 305-315.
98. Acosta-Vargas P., Salvador-Ullauri L., Jadán-Guerrero J. [и др.] Accessibility Assessment in Mobile Applications for Android // Advances in Human Factors and Ergonomics 2020: Proceedings of the AHFE 2020 International

Conference on Human Factors and Systems Interaction, July 16-20, 2020, Washington, DC, USA. - Cham: Springer, 2020. - Vol. 959. - P. 279-288.

99. Choi H., Park J. W. Methods to Improve Mobile Application Accessibility through Applying a Tactile User Interface in a Smartphone // Information (International Information Institute). - 2016. - Vol. 19, No 12. - P. 5327-5335.

100. Acosta-Vargas P., Salvador-Acosta B., Salvador-Ullauri L. (Dataset) Accessibility in Mobile Applications: Scoping Review. In Mendeley Data, Version. – 2021.

101. Teixeira P., Eusébio C., Teixeira L. Understanding the Integration of Accessibility Requirements in the Development Process of Information Systems: A Systematic Literature Review // Requirements Engineering. - 2024. - Vol. 29 - P. 1-34. - DOI: 10.1007/s00766-023-00409-8.

102. Bi T., Xia X., Lo D., Grundy J., Zimmermann T., Ford D. Accessibility in Software Practice: A Practitioner's Perspective // ACM Transactions on Software Engineering and Methodology. - 2022. - Vol. 31, No 4. - DOI: 10.1145/3503508.

103. Martin A. J., Strnadová I., Loblinzk J., Danker J. C., Cumming T. The role of mobile technology in promoting social inclusion among adults with intellectual disabilities // Journal of Applied Research in Intellectual Disabilities. - 2021. - Vol. 34, No 3. - P. 840-851. - DOI: 10.1111/jar.12869.

104. Korczak P., Zwierzchowska A. Using Mobile Applications in the Process of Enhancing and Restoring Abilities in Individuals with Intellectual Disabilities and Other Disabilities: A Literature Review // Advances in Rehabilitation. - 2020. - Vol. 34, No 4. - P. 36-41. - DOI: 10.5114/areh.2020.100774.

105. Al-Qahtani H. H. A., Al-Juda M. Q. Development of Social Skills with an Intellectual Disability Using Mobile Application // American Journal of Educational Research. [Электронный ресурс] – Режим доступа: <https://pubs.sciepub.com/education/6/1/9/index.html>.

106. Igual R., Plaza I., Martín L., Corbalan M., Medrano C. Guidelines to Design Smartphone Applications for People with Intellectual Disability: A Practical Experience // Ambient Intelligence - Software and Applications / eds. A. van Berlo et

al. - Cham: Springer, 2013. - Vol. 219 (Advances in Intelligent Systems and Computing). - P. 63-70. - DOI: 10.1007/978-3-319-00566-9_9.

107. Hardiyanti M., Pratomo D. N., Krisnandaru R. A. D., Riona V. Accessibility Evaluation of Interactive Learning Mobile Applications for Individuals with Intellectual Disabilities // International Journal of Information and Education Technology. - 2024. - Vol. 14, No 3. - P. 426-434. - DOI: 10.18178/ijiet.2024.14.3.2046.

108. Acosta-Vargas P., Salvador-Acosta B., Salvador-Ullauri L. [et al.] Accessibility in Native Mobile Applications for Users with Disabilities: A Scoping Review // Applied Sciences. - 2021. - Vol. 11, No 12. - Art. 5707. - DOI: 10.3390/app11125707.

109. SSilva G. M. S., Andrade R. M. C., Ticianne de Gois R. D. Design and Evaluation of Mobile Applications for People with Visual Impairments: A Compilation of Usable Accessibility Guidelines // Proceedings of the 18th Brazilian Symposium on Human Factors in Computing Systems (IHC 2019), Vitória, Brazil, October 22-25, 2019. - New York: ACM, 2019. - P. 1-10.

110. Pereira L. S., Matos M., Duarte C. Exploring Mobile Device Accessibility: Challenges, Insights, and Recommendations for Evaluation Methodologies // CHI '24: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, May 11-16, 2024, Honolulu, HI, USA. - New York: ACM, 2024. - P. 1-17. - DOI: 10.1145/3613904.3642526.

111. Patrick P., Obermeyer I., Xenakis J., Crocitto D., O'Hara D. Technology and social media use by adult patients with intellectual and/or developmental disabilities // Disability and Health Journal. - 2020. - Vol. 13, No 1. - DOI: 10.1016/j.dhjo.2019.100840.

112. Zhao Y., Gong L., Yang W., Zhou Y. How Accessibility Affects Other Quality Attributes of Software? A Case Study of GitHub // Science of Computer Programming. - 2024. - Vol. 231. - Art. 103027. - ISSN 0167-6423. - DOI: 10.1016/j.scico.2023.103027.

113. What is Accessibility in Mobile Apps and Why Does It Matter?
[Электронный ресурс] – Режим доступа:
<https://www.thedroidsonroids.com/blog/what-is-accessibility-in-mobile-apps>.

114. Peissner M., Schuller A., Spath D. A Design Patterns Approach to Adaptive User Interfaces for Users with Special Needs // Human-Computer Interaction. Design and Development Approaches / ed. J.A. Jacko. - Berlin, Heidelberg: Springer, 2011. - Vol. 6761 (Lecture Notes in Computer Science). - P. 264-273. - DOI: 10.1007/978-3-642-21602-2_30.

115. Liu Z. Interactive Interface Design for Adaptive and User Behavioural Habits // International Journal of Computer Science and Information Technology. - 2024. - Vol. 4, No 2. - P. 264-270. - DOI: 10.62051/ijcsit.v4n2.34.

116. Гурьев В. В., Шабалина О. А., Садовникова Н. П., Воронина А. А. Разработка модели жизненного цикла мобильных приложений для людей с ограниченными интеллектуальными возможностями // Программные продукты и системы. – 2023. – Т. 36, – № 4, – С. 573–581.

117. Сквозная разработка [Электронный ресурс] – Режим доступа:
<https://vivasoftltd.com/ru/end-to-end-development/>.

118. Kaspersky [Электронный ресурс] – Режим доступа:
<https://www.kaspersky.ru/>.

119. Принципы построения многомодульных мобильных приложений SurfStudio [Электронный ресурс] – Режим доступа:
<https://github.com/surfstudio/SurfAndroidStandard/blob/snapshot-%200.4.0/docs/common/multimodule/detail.md>.

120. Яндекс [Электронный ресурс] – Режим доступа:
<https://yandex.ru/company>.

121. Многомодульность в Android с точки зрения архитектуры. От А до Я [Электронный ресурс] – Режим доступа:
<https://m.habr.com/company/kaspersky/blog/422555/>.

122. Guryev V.V., Kosyakov S.V., Skvaznikov D.E., Shabalina O.A. Web Adaptation System for Mobile Application Interfaces for People with Intellectual

Disabilities // Information Technologies in Science, Management, Social Sphere and Medicine. – 2018. – P. 187-191.

123. Что такое JSON [Электронный ресурс] – Режим доступа: <https://habr.com/ru/articles/554274/>.

124. Smartphone Use and Activities by People with Disabilities [Электронный ресурс] – Режим доступа: <https://www.inclusivecitymaker.com/infographic-smartphone-use-disabled-people/>.

125. Accessibility in Native Mobile Applications for Users with Disabilities: A Scoping Review [Электронный ресурс] – Режим доступа: <https://www.mdpi.com/2076-3417/11/12/5707>.

126. Mobile Accessibility at W3C [Электронный ресурс] – Режим доступа: <https://www.w3.org/WAI/standards-guidelines/mobile/>.

127. 95% мобильных приложений не адаптированы для людей с ограниченными возможностями здоровья [Электронный ресурс] – Режим доступа: <https://roskachestvo.gov.ru/news/95-mobilnykh-prilozheniy-ne-adaptirovany-dlya-lyudey-s-ogranichennymi-vozmozhnostyami-zdorovya/>.

128. ISO/IEC 14102:2008 [Электронный ресурс] – Режим доступа: <https://www.iso.org/ru/standard/43189.html>.

129. Kravets A., Shcherbakov M., Kultsova M., Shabalina O. (Eds.) Design of Mobile Applications for People with Intellectual Disabilities // Creativity in Intelligent Technologies and Data Science: Proceedings. - Cham: Springer, 2015. - Vol. 535 (Communications in Computer and Information Science). - DOI: 10.1007/978-3-319-23766-4_65.

130. Shrestha S., Poudel P., Adhikari S., Adhikari I. Adaptive Menu: A Review of Adaptive User Interface // Trends in Computer Science and Information Technology. - 2022. - Vol. 7, No 3. - P. 103-106. - DOI: 10.17352/tcsit.000059.

131. Гурьев В. В., Шабалина О. А., Садовникова Н. П., Воронина А. А., Косяков С. В., Дмитриев Н. М. Технология сквозной разработки мобильных приложений для людей с ограниченными интеллектуальными возможностями // Онтология проектирования. — 2024. — Т. 14, № 2(52). — С. 230-242.

132. Igual R., Plaza I., Martín L., Corbalan M., Medrano C. Guidelines to Design Smartphone Applications for People with Intellectual Disability: A Practical Experience // Ambient Intelligence - Software and Applications / eds. A. van Berlo et al. — Heidelberg: Springer, 2013. — Vol. 219 (Advances in Intelligent Systems and Computing). — P. 65-69. — DOI: 10.1007/978-3-319-00566-9_9.

133. Mauro J., Marcelo G., Alejandro R. An AAC Mobile-Based Application for People with Intellectual Disability: A Case Study in Brazil // Advances in Human-Computer Interaction. — 2020. — P. 1-12. — DOI: 10.1155/2020/8932707.

134. What is Apache FreeMarker? [Электронный ресурс] – Режим доступа: <https://freemarker.apache.org/>.

135. Amvera Cloud [Электронный ресурс] – Режим доступа: <https://amvera.ru/>.

136. Shabalina O., Guriev V., Kosyakov S., Dmitriev N., Davtian A. MADM System for the Development of Adaptable Mobile Applications for People with Intellectual Disabilities // 2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA): Proceedings, Piraeus, Greece, July 15-17, 2020. - Piscataway: IEEE, 2020. - P. 1-8. - DOI: 10.1109/IISA50023.2020.9284409.

137. Soy Cappaz Mobile Application [Электронный ресурс] – Режим доступа: https://play.google.com/store/apps/details?id=com.mapfre.soycappaz&hl=es_AR.

138. ГБССУ СО ГПВИ "Волгоградский психоневрологический интернат [Электронный ресурс] – Режим доступа: <https://442fz.volganet.ru/025001/>.

139. MPI Oosterlo [Электронный ресурс] – Режим доступа: <https://www.mpi-oosterlo.be/>.

Приложение А. Результаты тестирования МП

Тестирование было проведено с сотрудниками и клиентами MPI Oosterlo

1. Experimental group: 2 users with ID

Name	Age	Sex	ID
Person 1	34y	man	Down syndrome
Person 2	36y	man	Down syndrome

2. Coaches:

2 staff of MPI Oosterlo

3. Event:

Social Ingenious – about user center design thinking

1) Результаты тестирования МП “Money Game”

User 1

Step	Problem
Choosing goods	Input price too hard, too small goods pictures,
Payment	Bad understand what to do, the first he tried to drag and drop coins from top to bottom

User 2

Step	Problem
Collecting coins in a wallet and other screens	1. Too small coins and paper. 2. He sees difference between 5 cents and 50 cents very bad or can't see in interface 3. He recognized coins 5 cents and 50 cents very bad 4. Didn't understand what him needs to collect real coins to wallet
Choosing goods	He couldn't input price
Payment	Too dark coins and small, he couldn't see them

User 1

Step	Problem
Accept category	He liked some pictures from mosaic but not all the images, he was confused about it

User 2

Step	Problem
Authorization	Too difficult authorization, too much symbols in login and password

Staff comments:

-I think, we need to test our designs constantly

-I think, we need to use cross-platform solutions for mobile app

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2023618672

Система поддержки разработки и эксплуатации
мобильных приложений

Правообладатель: *Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Волгоградский государственный технический
университет» (RU)*

Авторы: *Садовникова Наталья Петровна (RU), Шабалина
Ольга Аркадьевна (RU), Гурьев Владислав Витальевич
(RU), Воронина Ангелина Андреевна (RU)*



Заявка № 2023617796

Дата поступления 27 апреля 2023 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 27 апреля 2023 г.

Руководитель Федеральной службы
по интеллектуальной собственности

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 68b80077614c40f0a94edbd24145d5c7
Владелец **Зубов Юрий Сергеевич**
Действителен с 26.03.2022 по 26.05.2023

Ю.С. Зубов

Приложение В. Акт внедрения

А К Т

об использовании результатов кандидатской диссертации

Гурьева Владислава Витальевича

на тему «Автоматизация проектирования мобильных приложений для людей с ограниченными интеллектуальными возможностями»

Настоящий акт составлен о том, что веб-система, разработанная в рамках диссертационной работе Гурьева В.В., используется в компании ООО «Окели Энерджи» для разработки мобильных приложений с адаптируемым интерфейсом. Проектные и технологические решения, реализованные в системе, обеспечивают адаптацию интерфейсов мобильных приложений под конечных пользователей, и повышают таким образом их конкурентоспособность на рынке ПО.

Метод автоматизированного проектирования интерфейсов МП на основе адаптируемых шаблонов, автоматически генерируемых с помощью программно-реализуемой конфигурационной панели, обеспечивает возможность структурной и параметрической адаптации интерфейсов МП под конечного пользователя на этапе эксплуатации без изменения программного кода МП. Шаблон проектирования МП на основе многомодульной архитектуры со слойным делением модулей позволяет сократить количество модулей в архитектуре МП и снизить трудоемкость проектирования МП.

Применение веб-системы для разработки мобильных приложений с адаптируемым интерфейсом снижает трудоёмкость их разработки и повышает качество самих мобильных приложений.

Генеральный директор ООО
«Исследования и разработки «Окели
Энерджи»

Тюков Антон Павлович

