

Иблиев И.М.

Студент

Научный руководитель: Минкин А.В., к.ф.-м.н

Елабужский институт Казанский федеральный университет

Россия, г. Елабуга

ВЫБОР МЕТОДОЛОГИИ ДЛЯ ФУНКЦИОНАЛЬНО-СТРУКТУРНОГО АНАЛИЗА ПРИЛОЖЕНИЯ «PIZZA»

Аннотация: в статье рассматриваются различные методологии, применимые для функционально-структурного анализа мобильного приложения «Pizza». Подробно анализируются такие подходы, как UML для визуализации функций и взаимодействий, Agile для гибкой разработки с обратной связью, SWOT-анализ для оценки текущих позиций на рынке, а также User-Centered Design для создания пользовательских интерфейсов. В каждом случае акцентируется внимание на важности выбора правильной методологии в зависимости от специфики проекта и потребностей пользователей, что непосредственно влияет на успешность разработки приложения.

Ключевые слова: мобильные приложения, use-case диаграмма, функциональные возможности, проектирование, пользовательские группы, взаимодействие.

Ibliev I.M.

student

Scientific supervisor: Minkin A.V., PhD

Yelabuga Institute, Kazan Federal University

Russia, Yelabuga

CHOOSING A METHODOLOGY FOR FUNCTIONAL AND STRUCTURAL ANALYSIS OF THE PIZZA APPLICATION

Abstract: the article discusses various methodologies applicable to the functional and structural analysis of the Pizza mobile application. Approaches such as UML

for visualizing functions and interactions, Agile for agile development with feedback, SWOT analysis for evaluating current market positions, and User-Centered Design for creating user interfaces are analyzed in detail. In each case, attention is focused on the importance of choosing the right methodology, depending on the specifics of the project and the needs of users, which directly affects the success of application development.

Keywords: mobile applications, use-case diagram, functionality, design, user groups, interaction.

Функционально-структурный анализ является важным этапом при построении мобильного приложения и для его осуществления используется несколько методологий. Рассмотрим наиболее распространённые методологии которые можно применить для рассмотрения данной задачи [1].

UML — это стандартный язык моделирования, который позволяет визуализировать, специфицировать, конструировать и документировать программные системы. Выделяют следующие виды UML диаграмм. Use Case Diagrams, они определяют основные функции приложения и взаимодействие пользователя с системой. Class Diagrams моделируют структуру данных и взаимосвязи между объектами. Sequence Diagrams описывают взаимодействие между компонентами приложения в процессе выполнения функций. Позволяют четко визуализировать функциональность и структуру приложения. Упрощает коммуникацию между членами команды.

Agile — это гибкий подход к разработке программного обеспечения, который акцентирует внимание на итеративном процессе и постоянной обратной связи. Среди основных ее функций выделяют проведение спринтов для разработки и тестирования отдельных функций приложения проведение регулярных встреч с командой для обсуждения прогресса и выявления проблем, способность быстро адаптироваться к изменениям в требованиях, способность к активному вовлечению пользователей в процесс разработки.

SWOT-анализ (Strengths— сильные стороны, Weaknesses— слабые стороны, Opportunities — возможности, Threats — угрозы) позволяет оценить

сильные и слабые стороны приложения, а также возможности и угрозы на рынке. В этом случае выделяют такие особенности, как определение сильных сторон (например, уникальные функции или удобный интерфейс), выявляют слабые стороны (например, недостаток определенных функций), анализ возможностей для улучшения (например, расширение функционала), оценка угроз (например, конкуренция со стороны других приложений), помогает получить общее представление о позиции приложения на рынке, упрощает стратегическое планирование.

User-Centered Design (UCD) — это подход к проектированию, который ставит пользователя в центр процесса разработки. В этом случае проводится исследование пользователей для понимания их потребностей и предпочтений. Создаются прототипы интерфейса на основе полученных данных. Тестируются прототипы на реальных пользователях для получения обратной связи и стремятся обеспечить высокую степень удовлетворенности пользователей. Данная методология позволяет выявить проблемы на ранних этапах разработки.

Business Model Canvas — это инструмент для визуализации бизнес-модели приложения. Главными элементами являются определение ключевых партнеров, видов деятельности, ресурсов и каналов сбыта. Проводится анализ структуры затрат и источников дохода. Помогает понять бизнес-потенциал приложения. Упрощает стратегическое планирование и принятие решений.

Методология SADT (Structured Analysis and Design Technique) позволяет детализировать бизнес-процессы и функции системы, создавая структурированные диаграммы. Она хорошо подходит для создания предметной области и визуализации функций приложения. Простота понимания, возможность визуального представления с использованием диаграмм. Однако, она может быть менее эффективной для сложных систем с большим количеством взаимодействий.

IDEF0 (Integrated DEFinition Functions) является мощным инструментом для создания функционально-структурных диаграмм. Методология основана на создании блок-схем, отображающих функции системы, их связи и взаимодействие с внешними источниками. Она имеет четкую структуру, хорошо подходит для анализа и проектирования сложных систем, но может требовать больше времени для создания и понимания, особенно для новых пользователей.

Методология DFD (Data Flow Diagrams) используется для визуализации потоков данных внутри системы и взаимодействия между различными функциями. Эта методология позволяет понять, как данные перемещаются между различными компонентами приложения. Она фокусируется на потоках данных и взаимосвязях, но может не охватывать все аспекты функциональности приложения.

Выбор методологии для функционально-структурного анализа приложения "Пицца" должен основываться на конкретных целях анализа [2]. Например, если цель состоит в том, чтобы понять взаимодействие пользователей с приложением, то UML или UCD будут наиболее подходящими. Если же акцент делается на бизнес-потенциале или конкурентоспособности, то SWOT-анализ или Business Model Canvas могут быть более уместными.

Для разрабатываемого нами продукта остановим свой выбор на диаграмме UML, которая в значительной степени облегчает написание кода, графически описывает структуру проекта, взаимосвязь между функциональными элементами, позволяет взглянуть на систему в определенный момент ее выполнения. Она является очень хорошей презентацией внутренней структуры продукта [3, 4].

Использованные источники:

1. Забродин А. В. Методы и средства проектирования информационных систем: практикум : учебное пособие / А. В. Забродин, Д. А. Малунова. — Санкт-Петербург: ПГУПС, 2022. — 91 с.

2. Турнецкая Е. Л. Программная инженерия. Интеграционный подход к разработке / Е. Л. Турнецкая, А. В. Аграновский. — Санкт-Петербург: Лань, 2023. — 216 с.

3. Флегонтов А. В. Моделирование информационных систем. Unified Modeling Language / А. В. Флегонтов, И. Ю. Матюшичев. — 3-е изд., доп. — Санкт-Петербург: Лань, 2023. — 140 с.

4. Вейцман В. М. Проектирование информационных систем: учебное пособие для вузов / В. М. Вейцман. — 2-е изд., стер. — Санкт-Петербург: Лань, 2022. — 316 с.