**ATAM сессия для учёта НФТ в архитектуре системы**

**Задача**: реализация механизмов упрощенного внедрения новых продуктов в сети межконтинентальных закусочных “Замысловатость”.

Ключевые участники ATAM-сессии:

| **Роль участника** | **Направление (подразделение)** | **Описание** | **Список вопросов к группе респондентов** |
| --- | --- | --- | --- |
| Технолог | Бизнес | Ключевой бизнес-пользователь системы | Какие функции системы вам наиболее важны для выполнения вашей работы?  Какие проблемы вы сталкиваетесь с текущими системами, которые вы хотите решить с помощью новой системы?  Как вы оцениваете удобство использования интерфейса системы?  Какие данные и отчеты вам необходимы для эффективной работы?  Как вы видите процесс взаимодействия с другими пользователями системы? |
| Архитектор | ИТ-подразделение, архитектурный штаб | Архитектор отвечает за общую архитектуру системы и ее структуру. Его участие в сессии поможет определить, как различные компоненты системы будут взаимодействовать друг с другом, а также как обеспечить гибкость и расширяемость архитектуры. Архитектор может предложить решения для обеспечения надежности и устойчивости системы, а также помочь в выявлении потенциальных архитектурных рисков. | Как будет обеспечиваться масштабируемость системы? |
| Специалист по информационной безопасности | ИТ-подразделение, отдел информационной безопасности | Специалист по информационной безопасности отвечает за защиту данных и систем от угроз. Его знания о современных угрозах и методах защиты помогут определить необходимые меры безопасности, которые должны быть внедрены в систему. Он может предложить подходы к шифрованию данных, аутентификации пользователей и управлению доступом, что критически важно для обеспечения безопасности системы. | Какие меры безопасности необходимо внедрить для защиты данных?  Как будет осуществляться управление доступом к системе? |
| Специалист по сетевых доступам | ИТ-подразделение, отдел системного администрирования | Специалист по сетевым доступам отвечает за управление сетевой инфраструктурой и доступом к системе. Его участие поможет определить требования к сетевой безопасности, производительности и доступности. Он может предложить решения для обеспечения надежного и безопасного доступа к системе, а также помочь в выявлении потенциальных проблем с производительностью сети. | Как будет организовано сетевое взаимодействие между компонентами системы?  Как будет обеспечиваться доступ к системе для удаленных пользователей? |
| Руководитель проекта | Бизнес, отдел реализации проектов | Руководитель проекта отвечает за общее управление проектом, включая планирование, координацию и контроль за выполнением задач. Его участие в сессии поможет обеспечить, что все обсуждаемые качественные атрибуты соответствуют бизнес-целям и временным рамкам проекта. Он также может помочь в определении приоритетов и ресурсов, необходимых для реализации системы. | Каковы временные рамки и ключевые этапы внедрения системы?  Какие риски вы видите в процессе реализации проекта?  Как будет организовано взаимодействие между командами? |
| TeamLead со стороны разработки | ИТ-подразделение, отдел разработки программного обеспечения | TeamLead со стороны разработки отвечает за техническую реализацию проекта и управление командой разработчиков. Его знания о возможностях и ограничениях технологий, используемых в проекте, помогут определить, какие качественные атрибуты можно реализовать на практике. Он также может предложить идеи по оптимизации разработки и тестирования системы. | Какие технологии и инструменты вы планируете использовать для реализации системы?  Как будет организован процесс документирования кода и архитектуры системы?  Как вы видите процесс внедрения и развертывания системы в рабочую среду? |
| TeamLead со стороны анализа | ИТ-подразделение, отдел анализа | TeamLead со стороны анализа отвечает за сбор и анализ требований к системе. Его участие поможет обеспечить, что все бизнес-требования и ожидания пользователей будут учтены при определении качественных атрибутов. Он может помочь в формулировании требований и сценариев использования, а также в выявлении потенциальных проблем и рисков на ранних этапах разработки. | Как будет осуществляться сбор и анализ данных от пользователей?  Какие метрики будут использоваться для оценки успешности внедрения?  Как будет организовано взаимодействие с конечными пользователями для сбора обратной связи? |

**Качественные атрибуты системы**

1. **Производительность**: скорость обработки запросов и выполнения операций.
2. **Масштабируемость**: возможность увеличения нагрузки без потери качества работы.
3. **Надежность**: устойчивость к сбоям и возможность восстановления после них.
4. **Безопасность**: защита данных и предотвращение несанкционированного доступа.
5. **Удобство использования**: простота и интуитивность интерфейса для пользователей.

### Этапы сессии

1. **Подготовительный этап**:
   * Определение целей сессии.
   * Сбор предварительных данных о системе и ее требованиях.
   * Формирование списка участников и распределение по группам.
2. **Введение**:
   * Объяснение целей и задач ATAM сессии.
   * Обзор качественных атрибутов системы.
3. **Сбор требований**:
   * Каждая группа респондентов отвечает на подготовленные вопросы.
4. **Обсуждение и анализ**:
   * Обсуждение собранных ответов и выявление противоречий.
   * Оценка качественных атрибутов системы на основе полученных данных.
5. **Обсуждение компромиссов**:
   * Выявление возможных компромиссов между различными требованиями и ожиданиями.
   * Обсуждение, как можно сбалансировать интересы различных групп респондентов.
6. **Заключение**:
   * Подведение итогов сессии.
   * Формирование рекомендаций по дальнейшим действиям.
   * Определение ответственных за реализацию рекомендаций.

### Сценарии использования для оценки качества

1. **Оценка производительности при высокой нагрузке**
   * **Описание**: Система должна обрабатывать N запросов на сбор требований одновременно.
   * **Цель**: Измерить время отклика системы и её способность обрабатывать большое количество запросов без задержек.
2. **Тестирование надежности при сбоях**
   * **Описание**: Имитировать сбой в системе (например, отключение базы данных) и проверить, как система восстанавливается.
   * **Цель**: Оценить время восстановления и устойчивость системы к сбоям.
3. **Проверка безопасности доступа**
   * **Описание**: Попытаться получить доступ к системе с неправильными учетными данными и проверить, как система реагирует на несанкционированные попытки доступа.
   * **Цель**: Оценить механизмы аутентификации и авторизации, а также защиту данных.
4. **Тестирование удобства использования интерфейса**
   * **Описание**: Провести сессию с пользователями, чтобы они выполнили основные задачи (например, сбор требований, фиксация ответов) и оценили удобство интерфейса.
   * **Цель**: Собрать отзывы о навигации, интуитивности и доступности функций.
5. **Оценка масштабируемости системы**
   * **Описание**: Постепенно увеличивать количество пользователей и запросов к системе, чтобы определить, как она справляется с увеличением нагрузки.
   * **Цель**: Измерить, как система масштабируется и сохраняет производительность при увеличении числа пользователей.
6. **Проверка обработки ошибок**
   * **Описание**: Имитировать различные ошибки (например, неверные данные, сбой сети) и проверить, как система обрабатывает эти ситуации.
   * **Цель**: Оценить, насколько система устойчива к ошибкам и как она информирует пользователей о проблемах.
7. **Оценка времени выполнения задач**
   * **Описание**: Измерить время, необходимое пользователям для выполнения ключевых задач (например, сбор требований, внесение изменений).
   * **Цель**: Определить эффективность работы системы и выявить узкие места.

**Качественные атрибуты и их весовые коэффициенты**

|  |  |
| --- | --- |
| **Качественный атрибут** | **Весовой коэффициент** |
| Производительность | 0.25 |
| Масштабируемость | 0.10 |
| Надежность | 0.20 |
| Безопасность | 0.15 |
| Удобство использования | 0.20 |

**Оценка сценариев использования**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Сценарий использования** | **Производительность (1-5)** | **Масштабируемость (1-5)** | **Надежность (1-5)** | **Безопасность (1-5)** | **Удобство использования (1-5)** | **Интеграция (1-5)** | **Общая оценка** |
| 1. Оценка производительности при высокой нагрузке | 5 | 4 | 3 | 2 | 3 | 3 | 3.55 |
| 2. Тестирование надежности при сбоях | 3 | 3 | 5 | 2 | 3 | 3 | 3.35 |
| 3. Проверка безопасности доступа | 2 | 2 | 4 | 5 | 3 | 2 | 3.15 |
| 4. Тестирование удобства использования интерфейса | 3 | 3 | 3 | 2 | 5 | 3 | 3.35 |
| 5. Оценка масштабируемости системы | 4 | 5 | 3 | 2 | 3 | 3 | 3.85 |
| 6. Проверка обработки ошибок | 3 | 3 | 4 | 3 | 3 | 3 | 3.35 |
| 7. Оценка времени выполнения задач | 4 | 4 | 3 | 2 | 4 | 3 | 3.65 |

**Компромиссы:**

### Производительность vs. Надежность

* **Описание**: Для повышения производительности системы могут быть использованы оптимизации, такие как кэширование или асинхронная обработка, которые могут усложнить архитектуру и увеличить вероятность ошибок.
* **Компромисс**: Для достижения оптимального баланса между производительностью и надежностью можно использовать подходы, такие как внедрение механизмов мониторинга и автоматического восстановления. Это позволит системе быстро реагировать на сбои, сохраняя при этом высокую производительность. Например, можно использовать кэширование для повышения скорости обработки запросов, но при этом внедрить механизмы проверки целостности данных, чтобы минимизировать риски потери данных.

### Масштабируемость vs. Удобство использования

* **Описание**: Для обеспечения масштабируемости системы может потребоваться внедрение сложных архитектурных решений, таких как микросервисы, которые могут усложнить интерфейс и взаимодействие с пользователем.
* **Компромисс**: Для обеспечения масштабируемости системы без значительного ухудшения удобства использования можно применять адаптивные интерфейсы, которые подстраиваются под количество пользователей и их потребности. Это позволит сохранить простоту и интуитивность интерфейса, даже если архитектура системы становится более сложной. Например, можно использовать динамическую подгрузку контента, чтобы пользователи видели только необходимую информацию, что улучшит их опыт взаимодействия с системой.

### Безопасность vs. Удобство использования

* **Описание**: Для повышения безопасности системы могут быть внедрены строгие меры аутентификации и авторизации, которые могут усложнить процесс входа и использования системы.
* **Компромисс**: Для достижения баланса между безопасностью и удобством использования можно внедрить многофакторную аутентификацию, которая будет активироваться только в определенных ситуациях (например, при входе с нового устройства или из новой локации). Это позволит обеспечить высокий уровень безопасности, не перегружая пользователей лишними шагами в обычных условиях. Также можно использовать "умные" уведомления, которые информируют пользователей о подозрительных действиях, не требуя от них постоянного ввода дополнительных данных.

**Отчет о результатах ATAM-сессии для учета НФТ**

Данный отчет представляет результаты ATAM-сессии, проведенной для системы учета НФТ. Цель сессии заключалась в выявлении и оценке качественных атрибутов архитектуры системы, а также в принятии решений, направленных на оптимизацию ее функциональности и надежности.

**Список качественных атрибутов системы и их приоритеты**

|  |  |  |
| --- | --- | --- |
| **Качественный атрибут** | **Приоритет (1-5)** | **Обоснование приоритета** |
| Производительность | 5 | Высокая скорость обработки транзакций критична для пользователей. |
| Безопасность | 3 | Защита данных является базовыми требованиями. |
| Удобство использования | 5 | Интуитивный интерфейс необходим для продвижения среди пользователей и повышения их удовлетворенности. |
| Масштабируемость | 3 | Система должна поддерживать рост числа пользователей и транзакций. |
| Надежность | 3 | Система должна быть устойчива к сбоям, но не так критично, как производительность и удобство пользования. |

**Обсуждение и принятые решения:**

1. **Производительность**
   * **Решение**: Внедрить кэширование для ускорения обработки запросов и оптимизировать базу данных для повышения скорости транзакций. Также рассмотреть возможность использования асинхронной обработки для операций, не требующих немедленного ответа.
2. **Безопасность**
   * **Решение**: Реализовать многофакторную аутентификацию и шифрование данных на всех уровнях. Внедрить механизмы мониторинга для обнаружения подозрительных действий и автоматического реагирования на них.
3. **Удобство использования**
   * **Решение**: Разработать интуитивно понятный интерфейс с адаптивным дизайном, который будет подстраиваться под различные устройства. Провести тестирование с пользователями для получения обратной связи и улучшения интерфейса.
4. **Масштабируемость**
   * **Решение**: Использовать микросервисную архитектуру, которая позволит легко добавлять новые функции и масштабировать систему по мере роста числа пользователей. Также рассмотреть возможность использования облачных решений для динамического масштабирования ресурсов.
5. **Надежность**
   * **Решение**: Внедрить механизмы резервного копирования и восстановления данных, а также проводить регулярные тесты на устойчивость системы к сбоям.