

незаезда, если тот заблаговременно не отменил бронь и не воспользовался услугами гостиницы. Невозможно возместить потери при отсутствии заказов. Поэтому, если отели стремятся максимально увеличить свою прибыль, то им следует использовать все доступные методы продвижения своих услуг.

Одним из способов решения проблемы невосребованности услуг гостиничного комплекса является создание функционального сайта и его грамотное продвижение в сети Интернет. Большинство людей, пользующихся услугами различных отелей, являются продвинутыми интернет пользователями. Следовательно, для многих из них не возникает проблем при самостоятельном выборе отеля при помощи Интернета или просмотра сайтов тех отелей, которые предлагаются туристическими компаниями или специальными агентствами. По сути, сайт гостиницы и общение потенциального клиента с персоналом посредством телефона или при помощи иных способов связи, а также отзывы об отеле, являются единственными определяющими факторами при выборе того или иного отеля.

Любая гостиница, как и все остальные компании, нуждается в интернет ресурсе, на котором должен быть предоставлен информационный материал и контактная информация для сотрудников гостиницы и её потенциальных посетителей, а также сервисное обслуживание, позволяющее забронировать номер и предоставляющее справочную информацию об услугах гостиничного комплекса.

Функциональные возможности подсистемы позволяют формировать наличные и безналичные счета по тарифам гостиницы за предоставленные услуги, производить регистрацию, поселение, формировать заказы на поселение (бронирование номеров), вести контроль оказанных гостям услуг, вести учет телефонных переговоров, а также формировать и распечатывать необходимые отчеты по учету гостей и оплате оказанных услуг.

Основное назначение информационно-сопроводительного интернет-ресурса заключается в рекламировании услуг, предоставляемых гостиничным комплексом, бронировании его номеров посредством сети интернет с применением кредитных карт или других электронных средств оплаты при помощи технологий и коммуникационных решений, использующих интеллектуальные сетевые функции для поддержки высокого уровня доступности, безопасности и надежности.

Основными задачами подсистемы являются предоставление пользователям доступа к актуальной и достоверной в настоящий момент времени информации о гостинице, её деятельности, существующих проектах, акциях, о её новостях и событиях, оценке качества ее услуг, формирование счета за предоставленные услуги на основании введенной справочной информации и текущих оперативных данных по состоянию гостиничных номеров, контроль использования услуг гостями, а также получение необходимой внешней и внутренней отчетности.

Целью создания сайта является получение пользователями информации о компании, а также сокращение звонков в службу поддержки клиентов за счёт системы онлайн бронирования.

Целевая аудитория сайта – лица, возраст которых старше 18 лет.

В качестве языка и средства разработки была выбрана **Joomla** — свободное бесплатное программное обеспечение, распространяемое под лицензией GNU GPL. Система управления контентом (CMS) написана на языках PHP и Java Script. В качестве СУБД используется MySQL.

Информационная система, использующая СУБД, состоит из двух основных компонентов: сервера баз данных, управляющего данными и выполняющего поступающие от клиентских приложений запросы, и самих клиентских приложений, обеспечивающих интерфейс пользователя и посылающих запросы к серверу. Сервер баз данных манипулирует файлами, в которых хранятся данные, выполняет пользовательские запросы, поддерживает ссылочную целостность данных, обеспечивает доступ к ним, осуществляет резервное копирование данных и протоколирует операции, связанные с их изменением.

Список использованных источников:

1. Морозов, М. А. Информационные технологии в системах управления гостиничным комплексом / М. А. Морозов // Туризм: практика, проблемы, перспективы.
2. Тимохина, Т. Л. Технологии гостиничной деятельности // Теория и практика. Учебник / Т. Л. Тимохина — М., Юрайт, 2014 г..

ПРИМЕНЕНИЕ ОРТОГОНАЛЬНЫХ МАТРИЦ В ТЕСТИРОВАНИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Казюциц П.Ю.

Скудняков Ю. А. - канд.техн.наук, доцент

В настоящее время для проверки качества работы программного обеспечения используются различные подходы и технологии. В работе для решения данной задачи предлагается использовать метод тестирования ортогональными матрицами. Этот метод применим в случаях, когда у системы имеет место небольшая часть входных данных и большое число перестановок.

Использование ортогональных матриц позволяет проверить парные комбинации параметров и выявить оптимальные.

Результатами решения ортогональной матрицы в тестировании будет уменьшение количества пар составленных вариантов параметров и получение списка оптимальных пар для проверки работы приложения без потери охвата.

Метод тестирования с использованием ортогональной матрицы может быть осуществлен, например, при необходимости проверки работы приложения в нескольких браузерах (В) и операционных системах (S), полное количество пересечений в этом случае будет B^S . При этом каждая секция комбинаций для В во всех системах будет содержать некоторое число пересечений с другими комбинациями соседних секций. Решение такой ортогональной матрицы позволит избавиться от одинаковых и нерезультативных комбинаций, уменьшив число тестовых наборов, и выделить набор уникальных комбинаций, проверка которых наиболее приоритетна. Ортогональная матрица создается и рассчитывается программно на основе функции $F_x(a^y)$, где x – число строк, y – число столбцов (эквивалентное количеству входных параметров), a – варианты значений. Любые два столбца такой матрицы содержат все комбинации значений. Количество парных значений для этих двух столбцов всегда одинаково для всех вариантов пар (рисунок 1, рисунок 2).

На стадии подготовки данных для расчета матрицы определяются входные переменные, например, браузеры и операционные системы, и формируется блок принимаемых значений для каждой переменной. Построенная на основе этой информации ортогональная матрица будет содержать по столбцу для каждой переменной и строку как один тестовый случай. Вычисление матрицы выполняется программно на основе разработанного комбинаторного алгоритма. Следующим шагом после создания рабочей матрицы является финальная корректировка сгенерированной матрицы тестовых случаев: если в тестовом сценарии выделены приоритетные значения и переменные для проверки, то матрица будет программно отфильтрована. Оставшиеся пары значений будут воспроизведены как отдельные тестовые случаи. Использование ортогональных матриц гарантирует полное покрытие функционала программы, полученные пары значений считаются достаточно точными для качественной проверки.

Browser	Region	Device
IE	.CO.UK	Desktop
Firefox	.COM	iPad
Chrome	.CO.IN	Amazon Tab

Рисунок 1 – Список начальных значений пересечений

Browser	Region	Device
IE	.CO.UK	Desktop
IE	.COM	iPad
IE	.CO.IN	Amazon Tab
Firefox	.CO.UK	Desktop
Firefox	.COM	iPad
Firefox	.CO.IN	Amazon Tab
Chrome	.CO.UK	Desktop
Chrome	.COM	iPad
Chrome	.CO.IN	Amazon Tab

Рисунок 2 - Результат решения матрицы значений

Метод тестирования с использованием ортогональной матрицы наиболее эффективен на поздних этапах тестирования и рекомендуется к вводу на стабильно работающем приложении после отработки основных сценариев. Сценарий тестирования содержит неизменный файл с таблицами всех комбинаций параметров. На основе поступающего динамического файла с актуальными фильтрами и условиями проверки, соответствующими конкретному запуску, сценарий генерирует и рассчитывает матрицу, соотнося ее с указанными в динамическом файле условиями. Например, файл может содержать выборку браузеров, а не все доступные из списка. Изменяемый файл с критериями позволил нам задавать и корректировать будущий характер проверок еще до запуска тестового сценария. Результат решения матрицы в нашем сценарии сохраняется отдельным файлом и рассматривается далее как схема организаций запуска. Схема считывается программно, как дальнейшее руководство к созданию среды для тестирования. Таким образом, необходимость разработки тестовых программ человеком сведена к минимуму, так как сценарий использует начальную таблицу как масштаб работы, а динамический файл как уникальную корректировку, и для осуществления конкретного запуска генерирует актуальные тестовые варианты.