

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
Петрозаводский государственный университет
Математический факультет
Кафедра информатики и математического обеспечения

Отчет по дисциплине «Верификация ПО»

Выполнила:

Студентка 6 курса группы 22 608

К.О. Бохан

подпись

Преподаватель:

к.ф-м.н., доцент К. А. Кулаков

подпись

Петрозаводск

2014

Оглавление

1. Описание приложения	3
1.1 Функции приложения.....	4
2 Объект и стратегия тестирования	5
2.2 Стратегия тестирования.....	6
3. Детальный план тестов	12
3.1 Блочное тестирование	12
3.2 Интеграционное тестирование.....	21
3.3 Аттестационное тестирование	25
3.4 Тестирование безопасности.....	28
3.5 Покрытие кода тестами.....	32
3.6 Пример реализации тестов	32
4. Результаты тестирования.....	33
4.1 Сводная таблица по тестированию	33
4.2 Примеры найденных ошибок	34

1. Описание приложения

Лесозаготовительная промышленность является основой экономики Карелии. Однако отдача от единицы площади лесных угодий в Карелии в разы меньше, чем в странах Скандинавии. Для увеличения эффективности лесного комплекса необходимо использовать не только опыт соседних стран, но и внедрять современные информационные технологии в процесс заготовки древесины. Одной из задач, нуждающихся в автоматизации, является задача составления документации, а именно документов «Технологические карты». С целью решения этой задачи создается система, внедрение которой будет способствовать повышению эффективности предприятий лесного комплекса за счёт повышения производительности труда и качества проведения, основных и подготовительных лесосечных работ, сокращения времени прохождения и согласования технологической документации.

Система разрабатывается в виде насыщенного веб-приложения в связи с рядом преимуществ этого типа приложений, а именно: приложение не требует установки; обновление версий происходит автоматически; пользователи могут использовать приложение на любом устройстве, имеющем соединение с Интернетом, причем неважно, какая операционная система на нём установлена. Приложение включает клиентскую и сервисную части, для клиентской используются такие технологии, как HTML5, JavaScript, CSS, для сервисной – PHP, MySQL, асинхронное взаимодействие между частями возможно благодаря использованию AJAX подхода и библиотеки jQuery

1.1 Функции приложения

Основной функцией приложения является поддержка документооборота Технологических карт, под которой понимается наличие у системы функционала, позволяющего создавать электронные документы, редактировать их, а также отслеживать статусы документов.

В системе выделяются 2 типа пользователей:

- пользователь,
- менеджер.

Документ в системе находится в одном из 3-х статусов:

- формируется,
- отправлен на проверку,
- подтвержден.

Пользователи могут: создавать документы и редактировать их, отправлять документы на проверку менеджеру, просматривать проверяемые и подтвержденные документы. Пользователь имеет доступ только к созданным им документам.

Каждый пользователь закреплен за одним менеджером. Это означает предоставление доступа к документам пользователя менеджеру: менеджер может просматривать доступные документы, вносить изменения в документы, отправленные на проверку, изменять статус этих документов. Проверяемые документы могут быть подтверждены (что означает дальнейший запрет на редактирование и изменение статуса документа на «подтвержден») или возвращены пользователю для внесения изменений (изменение статуса документа на «формируется»). Менеджер имеет доступ только к документам закрепленным за ним пользователей.

Таким образом, в системе можно выделить следующий ряд функций:

- Аутентификация пользователя,
- Создание, просмотр и редактирование документов,
- Изменение статуса документа,

- Предоставление списка документов.

2 Объект и стратегия тестирования

Тестируемое веб-приложение можно разбить на 3 модуля: пользовательский интерфейс, адаптер, сетевой менеджер, сервис. Взаимодействие данных модулей отображено на рис. 1.

В качестве объекта блочного и интеграционного тестирования выбран модуль Адаптер. Объектом аттестационного тестирования служит приложение в целом.

Пользовательский интерфейс и сервисные скрипты являются протестированными продуктами сторонних разработчиков, в связи с чем проходят регрессионное тестирование в рамках описанного в плане аттестационного тестирования.

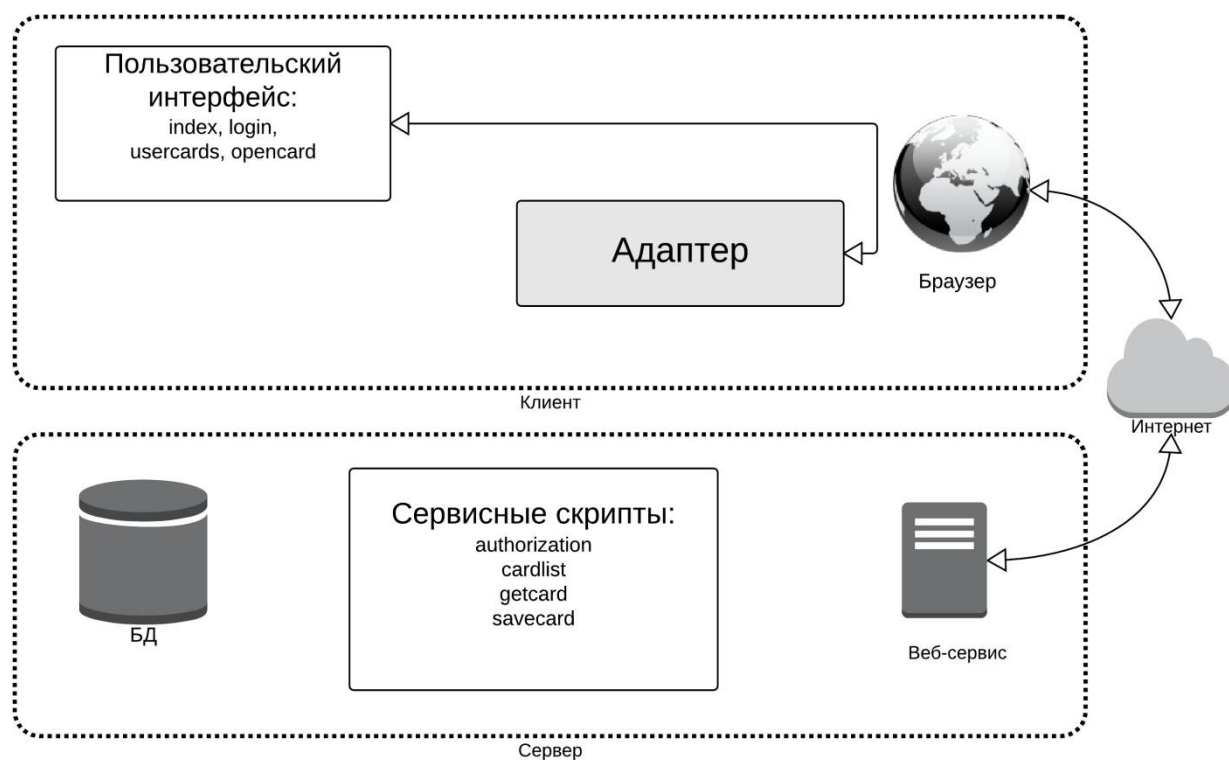


Рисунок 1 - Высокоуровневая архитектура

2.2 Стратегия тестирования

Блочное тестирование.

Объектом блочного тестирования выбран модуль Адаптер. Подробное описание сценариев данного модуля доступно в разделе 2.1.

Для проведения блочного тестирования используется библиотека QUnit. При тестировании сценариев с помощью этой библиотеки для каждого теста необходима html-страница, с подключенными к ней: библиотекой QUnit, тестируемым сценарием, заглушками вызываемых сценариев, а также самим тестовым сценарием (см. пример тестового сценария в разделе 3.5). Под заглушками понимаются специально созданные сценарии, поведение которых описано в тесте. Результаты тестирования будут доступны при открытии данной html-страницы в браузере.

Интеграционное тестирование.

Интеграционные тесты проверяют взаимодействие сценариев внутри модуля Адаптер.

Для проведения интеграционного тестирования применяется стратегия восходящего тестирования.

Порядок интеграции:

1. headerload() + ticketvalidation()
2. loginmessage() + ticketvalidation()
3. securityload() + ticketvalidation()
4. numpage () + ticketvalidation()
5. cardsload() + [numpage(), cardstemplate(), tablefooter()]
6. cardsave() + [filenamevalidation(), cardformvalidation()]

В связи со стратегией тестирования нет необходимости в использовании заглушек вызываемых сценариев. Это позволяет проводить тестирование непосредственно в браузере, например в Google Chrome с использованием Google Chrome Developers Tools. Также возможно использование библиотеки

QUnit аналогично способу, описанному в блочном тестировании, с подключением самих сценариев вместо заглушек.

Аттестационное тестирование.

Аттестационное тестирование проводится для всей системы, что подразумевает выполнение действий в пользовательском интерфейсе.

Для проведения тестирования в данном случае необходим браузер.

Тестирование безопасности

Под тестированием безопасности в данном плане понимается определение ролей и проверка списка функций системы, доступных для каждой роли. Включает проверку методов шифрования данных при хранении и передаче, отказа доступа к запрещенным функциям, документам, перехвата данных, подделки идентификатора сессии.

Функции системы, доступные для каждой из ролей, описаны в разделе 1.1.

Критерии остановки и возобновления тестирования

Тест считается успешно пройденным, если ожидаемый и фактический результаты совпадают. В противном случае производится заключение о найденной ошибке.

В связи со стратегией тестирования, тестирование сценариев более высокого уровня иерархии должно быть приостановлено при нахождении критических ошибок в вызываемых ими сценариях. В случаях возникновения существенных ошибок тестирование может быть продолжено на усмотрение тестировщика.

Работы по тестированию возобновляются после исправления ошибок, вызвавших приостановку тестирования.

Результаты тестирования должны быть представлены в январе 2015 года.

2.1 Подробное описание элементов системы

Взаимодействие страниц и сценариев, а также порядок их вызовов представлены на рис.2.

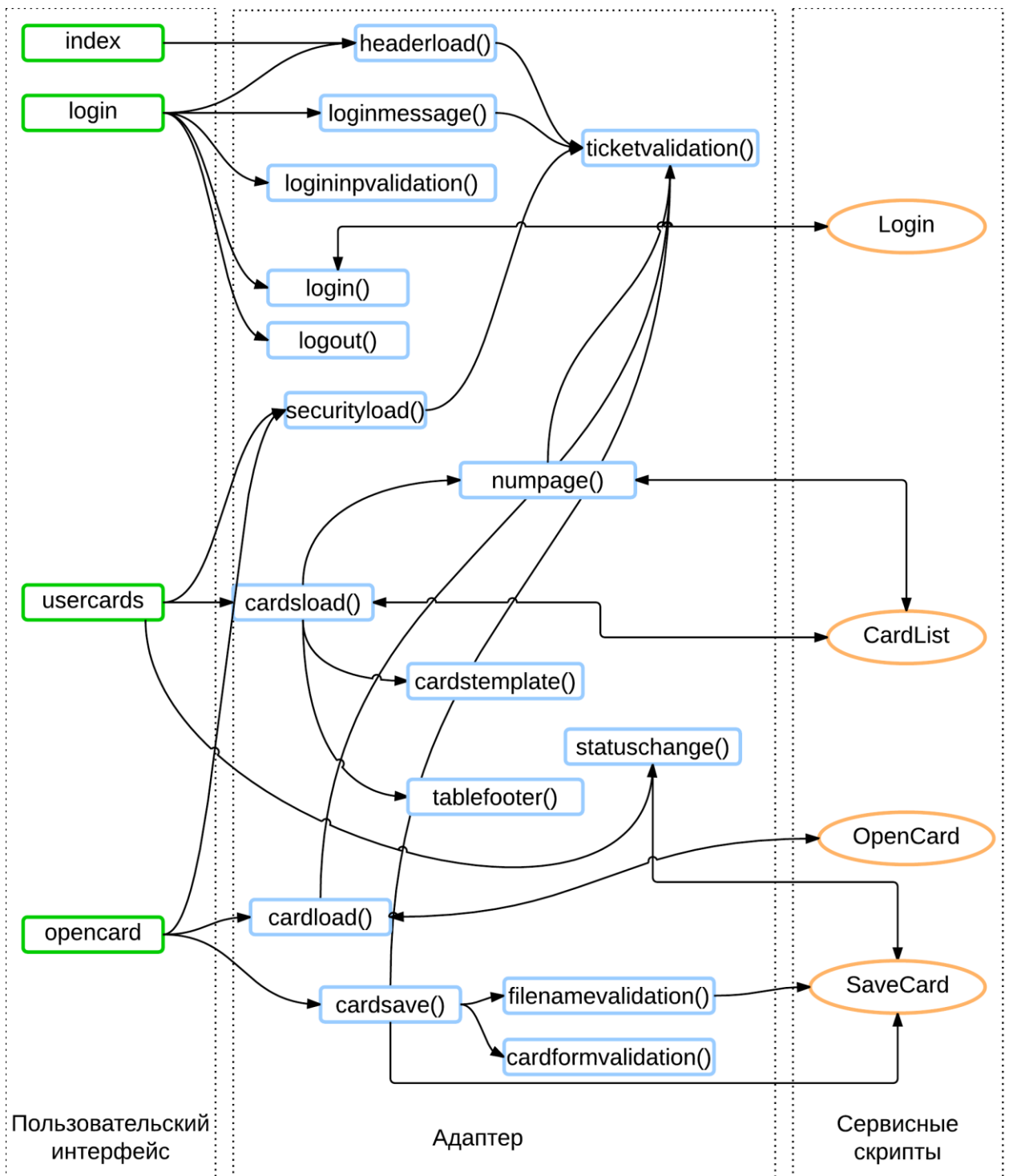


Рисунок 2 - Элементы системы

- **Пользовательский интерфейс:** *аттестационное тестирование*
 - Index – главная страница приложения, содержит краткую информацию о системе и ссылку на страницу входа в систему для неавторизованных пользователей или ссылку на страницу списка технологических карт и кнопку выхода из системы для авторизованных пользователей,
 - Usercards – страница, содержащая список документов авторизованного пользователя. Список выводится постранично и содержит краткую информацию о документах. Помимо списка Usercards содержит кнопки навигации по страницам (первая, предыдущая, текущая, следующая, последняя)
 - Login – страница для авторизации, содержит форму для ввода имени пользователя и пароля, а также информацию о пользователе, уже вошедшем в систему,
 - Open card – страница, содержащая форму с данными документа и кнопки «Сохранить» и «Завершить».
- **Адаптер:** *блочное, интеграционное, аттестационное тестирование*
 - Headerload – сценарий, изменяющий отображение шапки страниц в зависимости от авторизации пользователя,
 - Ticketvalidation – сценарий, проверяющий наличие идентификатора сессии и его соответствие принятому стандарту,
 - Loginmessage – сценарий, вносящий информацию о вошедшем в систему пользователе на странице Login,
 - Logininvalidation – сценарий, проверяющий на соответствие принятому стандарту данных, введенных в поля формы на странице Login,
 - Login – сценарий, производящий авторизацию пользователя на основе введенных в поля формы на странице Login данных и взаимодействия с сервисом,
 - Logout – сценарий, завершающий текущую сессию,

- Securityload – сценарий, производящий переадресацию на страницу Login в случае попытки доступа неавторизованным пользователем на страницу, доступную только авторизованным пользователям,
- Cardsload – сценарий, получающий в зависимости от параметров адресной строки и роли пользователя необходимую страницу списка документов авторизованного пользователя, и заполняющий шаблон краткой информации о документе
- Numpage – сценарий, возвращающий количество страниц списка, доступного для авторизованного пользователя
- Cardstemplate – шаблон краткой информации о документе, описывает отображение информации для пользователей разных ролей и действия, доступные для документа в зависимости от его статуса,
- Tablefooter – сценарий, вносящий информацию в кнопки навигации по страницам в Usercards (номер текущей страницы, видимость кнопок: скрывает недоступные),
- Statuschange – сценарий «быстрого» изменения статуса документа: подтверждение или откат документа без перехода на страницу просмотра документа. Доступен для менеджера.
- Cardload – сценарий, получающий с сервиса данные открываемого документа и заполняющий форму на странице Opencard
- Cardsave – сценарий, загружающий на сервис данные документа,
- Filenamevalidation – сценарий, проверяющий соответствие введенного названия документа принятым стандартам и доступность такого имени файла на сервисе,
- Cardformvalidation – сценарий, проверяющий на соответствие принятым стандартам данных, введенных в поля формы.

Сервисные сценарии: аттестационное тестирование

- Login – авторизация пользователя, выдача идентификатора сессии,
- CardList– получение страницы списка документов, для которых у авторизованного пользователя хватает прав на просмотр, предоставление информации об общем числе страниц списка,
- OpenCard – получение данных документа для просмотра или редактирования в зависимости от параметров сценария и наличия соответствующих прав,
- SaveCard – загрузка данных документа на сервис, проверка доступности имени файла.

Принятые стандарты:

№	Объект	Стандарт
1	ticket – идентификатор сессии	Содержит 32 байта
2	login – имя пользователя, pswd – пароль filename – название документа	Содержит от 3-х до 20-ти символов, допустимо содержание только символов кириллицы, латинского алфавита и цифр
3	role – роль пользователя	{ manager, user }
4	Данные о сессии	login, ticket, role
5	name = int* - числовые поля	содержат только положительные числа
6	name = txt* - текстовые поля	содержат только символы кириллицы и латинского алфавита, цифры, точки и пробелы
7	status – статус документа	{ newfile, checkfile, closedfile }
8	action – способ доступа к файлу	{ open, edit }

3. Детальный план тестов

3.1 Блочное тестирование

Сценарий: ticketvalidation()

Описание: проверка наличия идентификатора сессии в cookie и его соответствие принятому стандарту №1

№	Условия	Ожидаемый результат	Тип теста
1.1	Cookie содержит 'ticket' необходимой длины	true	общий
1.2	Cookie не содержит 'ticket'	false	негативный
1.3	Cookie содержит 'ticket' = null	false	негативный
1.4	Cookie содержит 'ticket' = ""	false	негативный
1.5	Cookie содержит 'ticket' длины 31 байт	false	негативный
1.6	Cookie содержит 'ticket' длины 33 байта	false	негативный

Сценарий: logininvalidation()

Описание: проверка соответствия введенных пользователем имени и пароля принятому стандарту №2

№	Условия	Ожидаемый результат	Тип теста
2.1	Ввод имени пользователя и пароля необходимой длины, содержащих только цифры и латинские символы	"disabled" = false	общий
2.2	Ввод имени пользователя длиной 1 символ	"disabled" = true Вывод сообщения о неверном числе символов имени пользователя	негативный
2.3	Ввод имени пользователя, длиной 2 символа	"disabled" = true Вывод сообщения о неверном числе символов имени пользователя	негативный
2.4	Ввод имени пользователя длиной 3 символа и пароля необходимой длины, содержащих только цифры и	" disabled" = false	краевой

	латинские символы		
2.5	Ввод имени пользователя длиной 20 символов и пароля необходимой длины, содержащих только цифры и латинские символы	"disabled"= false	краевой
2.6	Ввод пароля длиной 1 символ	"disabled"= true Вывод сообщения о неверном числе символов пароля	негативный
2.7	Ввод имени пользователя, длиной 2 символа	"disabled"= true Вывод сообщения о неверном числе символов пароля	негативный
2.8	Ввод имени пользователя необходимой длины и пароля из 3-х символов, содержащих только цифры и латинские символы	"disabled"= false	краевой
2.9	Ввод имени пользователя необходимой длины и пароля длиной 20 символов, содержащих только цифры и латинские символы	"disabled"= false	краевой
2.10	Ввод имени пользователя длиной 21 символ	"disabled"= true Вывод сообщения о неверном числе символов имени пользователя	негативный
2.11	Ввод пароля длиной 21 символ	"disabled"= true Вывод сообщения о неверном числе символов пароля	негативный
2.12	Ввод имени пользователя и пароля необходимой длины, содержащих не только цифры и латинские символы	"disabled"= true Вывод сообщения о допустимых символах	негативный

Сценарий: login(login, pswd)

Описание: авторизация пользователя: вызов сервисного сценария авторизации с параметрами: имя пользователя, логин; разбор ответа сервиса; в случае подтверждения авторизации: запись данных о сессии в cookie, иначе – вывод сообщения об ошибке

Зависимости: Login, используется заглушка

№	Условия	Ожидаемый результат	Тип теста
3.1	login, pswd содержат данные о зарегистрированном пользователе	Наличие данных о сессии в cookie, вызов переадресации на главную страницу приложения	общий
3.2	login, pswd содержат данные необходимой длины (от 3-х до 20-ти символов), пользователь не зарегистрирован в системе	errmsg.text: Пользователя нет в системе	негативный
3.3	login или pswd содержат данные, несоответствующей стандарту длины : 2 символа	Сценарий завершается без возникновения ошибки	негативный
3.4	login или pswd содержат данные, несоответствующей стандарту длины : 21 символ	Сценарий завершается без возникновения ошибки	негативный

Сценарий: logout()

Описание: завершение сессии на стороне клиента: удаление данных о сессии из cookie

№	Условия	Ожидаемый результат	Тип теста
4.1	cookie содержат данные о сессии	cookie не содержат данных о сессии	общий
4.2	cookie не содержат данных о сессии	cookie не содержат данных о сессии	специальный

Сценарий: headerload(login, logout)

Описание: изменение параметров отображения блоков в заголовке страницы, вызывающей сценарий

Зависимости: ticketvalidation(), используется заглушка

№	Условия	Ожидаемый результат	Тип теста
5.1	ticketvalidation(): true	logout.display: none	общий
5.2	ticketvalidation(): false	login.display: none	общий

Сценарий: loginmessage(errmessage)

Описание: вывод сообщения об авторизованном пользователе

Зависимости: ticketvalidation(), используется заглушка

№	Условия	Ожидаемый результат	Тип теста
5a.1	ticketvalidation(): true, cookie содержит имя пользователя	errmessage.text: вход в систему уже выполнен с указанием имени пользователя	общий
5a.2	ticketvalidation(): false	Сценарий завершается без возникновения ошибки	общий
5a.3	ticketvalidation(): true, cookie не содержит имя пользователя	Сценарий завершается без возникновения ошибки	негативный

Сценарий: securityload()

Описание: переадресация неавторизованных пользователей на страницу авторизации

Косвенные данные: loginpageURL

Зависимости: ticketvalidation(), используется заглушка

№	Условия	Ожидаемый результат	Тип теста
6.1	ticketvalidation(): true	Сценарий завершается без возникновения ошибки	общий
6.2	ticketvalidation(): false	Вызов переадресации на страницу авторизации loginpageURL	общий

Сценарий: numpage()

Описание: определение числа страниц списка документов пользователя: передача сервису идентификатора сессии, получение данных от сервиса, возврат полученных от сервиса данных или значения по умолчанию, если сервис данных не предоставил

Зависимости: ticketvalidation(), CardList, используются заглушки

№	Условия	Ожидаемый результат	Тип теста
7.1	ticketvalidation(): true, пользователю доступно page страниц	page	общий
7.2	ticketvalidation(): true, сервис недоступен	значение по умолчанию (50)	специальный
7.3	ticketvalidation(): false	Сценарий завершается	негативный

		без возникновения ошибки	
--	--	-----------------------------	--

Сценарий: statuschange(fid, newstatus)

Описание: изменение менеджером статуса документа с идентификатором fid на статус newstatus

Косвенные данные: errmsgmessage

Зависимости: ticketvalidation(), SaveCard, используются заглушки

№	Условия	Ожидаемый результат	Тип теста
8.1	Роль пользователя: менеджер, идентификатор содержит только цифры, newstatus = CLOSEDFILE или NEWFILE, ticketvalidation(): true	Вызов обновления страницы	общий
8.2	Роль пользователя: менеджер, идентификатор содержит не только цифры, newstatus = CLOSEDFILE или NEWFILE, ticketvalidation(): true	Запрос к сервису не осуществляется	негативный
8.3	Роль пользователя: менеджер, идентификатор содержит только цифры, newstatus = CHECKFILE, ticketvalidation(): true	Запрос к сервису не осуществляется	негативный
8.4	Роль пользователя: менеджер, идентификатор содержит только цифры, newstatus = CLOSEDFILE или NEWFILE, ticketvalidation(): false,	Запрос к сервису не осуществляется	негативный
8.5	Роль пользователя: менеджер, идентификатор содержит только цифры, newstatus = CLOSEDFILE или NEWFILE, ticketvalidation(): true, сервис недоступен	Errmessage.text: невозможно изменить статус	негативный
8.6	Роль: пользователь	Запрос к сервису не осуществляется	негативный

Сценарий: cardstemplate(role,documentinfo)

Описание: заполнение шаблонов отображения краткой информации о документе в зависимости от доступных пользователю действий по отношению к документам

№	Условия	Ожидаемый результат	Тип теста
9.1	role: user documentinfo.status: NEWFILE	Шаблон содержит кнопку перехода на страницу просмотра документа	общий
9.2	role: user documentinfo.status: CHECKFILE	Шаблон содержит элемент перехода на страницу просмотра документа	общий
9.3	role: user documentinfo.status: CLOSEDFILE	Шаблон содержит элемент перехода на страницу просмотра документа	общий
9.4	role: manager documentinfo.status: NEWFILE	Шаблон содержит элемент перехода на страницу просмотра документа	общий
9.5	role: manager documentinfo.status: CHECKFILE	Шаблон содержит элемент перехода на страницу просмотра документа	общий
9.6	role: manager documentinfo.status: CLOSEDFILE	Шаблон содержит элемент перехода на страницу просмотра документа	общий
9.7	Роль или статус отличаются от вышеперечисленных	Информация о документе не выводится	негативный

Сценарий: tablefooter(page, maxpage)

Описание: корректное отображение кнопок навигации по страницам списка: скрытие недоступных элементов навигации

Косвенные данные: footer

№	Условия	Ожидаемый результат	Тип теста
10.1	page:1 maxpage:1	footer.text: Текущая страница:1	краевой
10.2	page:1 maxpage:2	footer.text: Текущая страница:1 Следующая страница:2	краевой

10.3	page:1 maxpage:3	footer.text: Текущая страница:1 Следующая страница:2 Последняя страница:3	краевой
10.4	page:1 maxpage:10	footer.text: Текущая страница:1 Следующая страница:2 Последняя страница:10	краевой
10.5	page:2 maxpage:2	footer.text: Предыдущая страница: 1 Текущая страница:2	краевой
10.6	page:3 maxpage:3	footer.text: Первая страница: 1 Предыдущая страница: 2 Текущая страница:3	краевой
10.7	page:4 maxpage:10	footer.text: Первая страница: 1 Предыдущая страница: 2 Текущая страница:4 Следующая страница:5 Последняя страница:10	общий
10.8	page > maxpage	footer.display: none	негативный
10.9	Заданы отрицательные или нечисловые параметры	footer.display: none	негативный

Сценарий: cardsload()

Описание: загрузка страницы списка документов авторизованного пользователя, номер необходимой страницы указывается в параметре адресной строки

Косвенные данные: cookie

Зависимости: numpage(), cardstemplate(), tablefooter(), CardList, используются заглушки

№	Условия	Ожидаемый результат	Тип теста
11.1	ticketvalidation(): true, указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул краткую информацию о документах, содержащихся на странице списка, в cookie содержится	Вызов cardstemplate(), tablefooter(), передача в качестве параметров cardstemplate роли пользователя и возвращенного сервисом массива с информацией о документах	общий

	роль пользователя		
11.2	ticketvalidation(): false	запрос к сервису не осуществляется	негативный
11.3	ticketvalidation(): true, указан номер запрашиваемой страницы, находящийся вне диапазона от 1 до общего числа доступных страниц	запрос к сервису не осуществляется	негативный
11.4	ticketvalidation(): true, указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул сообщение об ошибке	errmsg.message.text: : невозможно загрузить список	негативный
11.5	ticketvalidation(): true, указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул краткую информацию о документах, содержащихся на странице списка, в cookie не содержится роль пользователя	Сценарий завершается без возникновения ошибки	негативный

Сценарий: cardformvalidation(inputs)

Описание: проверка соответствия введенных на форме данных принятым стандартам № 5 и №6

Косвенные данные: errmsg.message

№	Условия	Ожидаемый результат	Тип теста
12.1	Все поля заполнены в соответствии с описанным стандартом	Сценарий завершается без возникновения ошибки	общий
12.2	Несколько полей содержат недопустимые символы или не заполнены	errmsg.message.text: <количество неверно заполненных полей>, изменение класса этих полей на « wronginput»	негативный
12.3	Данные в полях с классом « wronginput» изменены на	Удаление класса на « wronginput» у	специальный

	правильные	обновленных полей	
--	------------	-------------------	--

Сценарий: filenamevalidation(filename)

Описание: проверка доступности и соответствия filename принятому стандарту №2

Косвенные данные: errmessage

№	Условия	Ожидаемый результат	Тип теста
13.1	filename содержит менее 3-х или более 20-ти символов	errmessage.text: : неверное название документа	негативный
13.2	filename содержит более 3-х и менее 20-ти символов, включающих недопустимые	errmessage.text: : неверное название документа	негативный
13.3	filename содержит более 3-х и менее 20-ти символов, включающих только допустимые, сервис подтверждает возможность использования имени	Сценарий завершается без возникновения ошибки	общий
13.4	filename содержит более 3-х и менее 20-ти символов, включающих только допустимые, сервис запрещает использования имени	errmessage.text: : невозможно сохранить на сервисе документ с таким именем	негативный

Сценарий: cardload()

Описание: получение от сервиса документа для с определенным идентификатором определенного действия (редактирование, просмотр).

Косвенные данные: Идентификатор и тип действия (заданы в параметрах адресной строки)

№	Условия	Ожидаемый результат	Тип теста
14.1	Идентификатор и тип действия не заданы	Сценарий завершается без возникновения ошибки	негативный
14.2	Задан нечисловой идентификатор или тип действия, отличающийся от OPEN и EDIT	Сценарий завершается без возникновения ошибки	негативный
14.3	Задан числовой идентификатор и тип	Поля формы документа заполнены отданными сервисом данными,	общий

	действия: OPEN, сервис отдал текст документа	документ недоступен для редактирования, кнопки сохранения и завершения не отображаются	
14.4	Задан числовой идентификатор и тип действия: EDIT, сервис отдал текст документа	Поля формы документа заполнены отданными сервисом данными, документ доступен для редактирования	общий
14.5	Задан числовой идентификатор и тип действия: EDIT, сервис вернул сообщение об ошибке	Поля формы документа заполнены отданными сервисом данными, документ доступен для редактирования	общий

Сценарий: cardsave()

Описание: создание нового или обновление ранее сохраненного файла на сервисе в зависимости от параметра адресной строки

Косвенные данные: usercardsURL

Зависимости: filenamevalidation(), cardformvalidation(), используются заглушки, всегда возвращающие истину

№	Условия	Ожидаемый результат	Тип теста
15.1	Идентификатор файла задан, сервис вернул подтверждение обновления	Переадресация на страницу со списком документов usercardsURL	общий
15.2	Идентификатор файла не задан, сервис вернул подтверждение сохранения	Переадресация на страницу со списком документов usercardsURL	общий
15.3	Идентификатор файла задан, сервис вернул ошибку	Вывод сообщения об ошибке	негативный
15.4	Идентификатор файла не задан, сервис вернул ошибку	Вывод сообщения об ошибке	негативный

3.2 Интеграционное тестирование

Сценарий: ticketvalidation() + headerload()

Описание: изменение параметров отображения блоков в заголовке страницы, вызывающей сценарий

№	Условия	Ожидаемый результат	Тип теста
---	---------	---------------------	-----------

16.1	cookie.ticket установлен	logout.display: none	общий
16.2	cookie.ticket не установлен	login.display: none	общий

Сценарий: ticketvalidation() + loginmessage()

Описание: вывод сообщения об авторизированном пользователе

№	Условия	Ожидаемый результат	Тип теста
17.1	cookie содержит имя пользователя и идентификатор сессии	Сообщение о том, что вход в систему уже выполнен с указанием имени пользователя	общий
17.2	cookie не содержит идентификатор сессии	Сценарий завершается без возникновения ошибки	общий
17.3	cookie не содержит имя пользователя	Сценарий завершается без возникновения ошибки	негативный

Сценарий: ticketvalidation() + securityload()

Описание: переадресация неавторизированных пользователей на страницу авторизации

№	Условия	Ожидаемый результат	Тип теста
18.1	cookie.ticket установлен	Сценарий завершается без возникновения ошибки	общий
18.2	cookie.ticket не установлен	Переадресация на страницу авторизации	общий

Сценарий: ticketvalidation() + numpage ()

Описание: определение числа страниц списка документов пользователя: передача сервису идентификатора сессии, получение данных от сервиса, возврат полученных от сервиса данных или значения по умолчанию если сервис данных не предоставил

№	Условия	Ожидаемый результат	Тип теста
19.1	cookie.ticket установлен, сервис вернул количество страниц = page	page	общий
19.2	cookie.ticket установлен, сервис не вернул количество страниц	значение по умолчанию (50)	специальный
19.3	cookie.ticket не установлен	Сценарий завершается	негативный

		без возникновения ошибки	
--	--	--------------------------	--

Сценарий: [numpage(), cardstemplate(), tablefooter()] + cardsload()

Описание: загрузка страницы списка документов авторизованного пользователя, номер необходимой страницы указывается в параметре адресной строки

№	Условия	Ожидаемый результат	Тип теста
20.1	cookie.ticket установлен , указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул краткую информацию о документах, содержащихся на странице списка, в cookie содержится роль пользователя	Вызов cardstemplate(), tablefooter(), передача в качестве параметров cardstemplate роли пользователя и массива с информацией о документах, возвращенной сервисом	общий
20.2	cookie.ticket не установлен	запрос к сервису не осуществляется	негативный
20.3	cookie.ticket установлен , указан номер запрашиваемой страницы, находящийся вне диапазона от 1 до общего числа доступных страниц	запрос к сервису не осуществляется	негативный
20.4	cookie.ticket установлен , указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул сообщение об ошибке	вывод сообщения об ошибке: невозможно загрузить список	негативный
20.5	cookie.ticket установлен , указан номер запрашиваемой страницы, находящийся в диапазоне от 1 до общего числа доступных страниц, сервис вернул краткую информацию о документах, содержащихся на странице списка, в cookie не содержится роль	Сценарий завершается без возникновения ошибки	негативный

	пользователя		
--	--------------	--	--

Сценарий: [filenamevalidation(), cardformvalidation()] + cardsave()

Описание: создание нового или обновление ранее сохраненного файла на сервисе в зависимости от параметра адресной строки

№	Условия	Ожидаемый результат	Тип теста
21.1	Идентификатор файла задан, сервис вернул подтверждение обновления	Переадресация на страницу со списком документов	общий
21.2	Идентификатор файла не задан, имя файла соответствует стандарту №2, данные формы документа соответствуют стандартам №5 и №6 сервис вернул подтверждение сохранения	Переадресация на страницу со списком документов	общий
21.3	Идентификатор файла не задан, имя файла не соответствует стандарту №2	Вывод сообщения об ошибке	негативный
21.4	Идентификатор файла не задан, имя файла соответствует стандарту №2, данные формы документа не соответствуют стандартам №5 и №6, сохранение подтверждено пользователем, сервис вернул подтверждение сохранения	Переадресация на страницу со списком документов	общий
21.5	Идентификатор файла не задан, имя файла соответствует стандарту №2, данные формы документа не соответствуют стандартам №5 и №6, сохранение не подтверждено пользователем	Сценарий завершается без возникновения ошибки	специальный

3.3 Аттестационное тестирование

22.1

Сценарий: Вход в систему

Тестируемая функциональность: Аутентификация

Описание: Ввод данных пользователя, зарегистрированного в системе

Тип теста: общий

Ожидаемый результат: Вход выполнен: переадресация на главную страницу приложения, в шапке сайта доступны: переход на страницу со списком документов и кнопка «Выход»

22.2

Сценарий: Вход в систему

Тестируемая функциональность: Аутентификация

Описание: Ввод данных пользователя, не зарегистрированного в системе

Тип теста: негативный

Ожидаемый результат: Вход не выполнен: на странице авторизации выведена ошибка, в шапке сайта переход на страницу со списком документов недоступен

22.3

Сценарий: Выход из системы

Тестируемая функциональность: Аутентификация

Описание: Нажатие кнопки «Выход»

Тип теста: общий

Ожидаемый результат: Выполнен выход из системы: переадресация на главную страницу приложения, в шапке сайта переход на страницу со списком документов недоступен

23.1

Сценарий: Просмотр списка документов: пользователь

Тестируемая функциональность: Предоставление списка документов

Описание: Переход на страницу со списком документов авторизованным пользователем

Тип теста: общий

Ожидаемый результат: На странице отображен первый лист списка документов пользователя, кнопки навигации по списку (в случае доступности нескольких листов списка), в зависимости от статуса документа доступны различные действия с документами (для незавершенных и неотправленных на проверку документов доступно редактирование, для остальных - просмотр)

23.2

Сценарий: Просмотр списка документов: пользователь

Тестируемая функциональность: Предоставление списка документов

Описание: Переход на страницу со списком документов неавторизованным пользователем

Тип теста: негативный

Ожидаемый результат: переадресация на страницу авторизации

23.3

Сценарий: Просмотр списка документов: менеджер

Тестируемая функциональность: Предоставление списка документов

Описание: Переход на страницу со списком документов авторизованным менеджером

Тип теста: общий

Ожидаемый результат: На странице отображен первый лист списка документов менеджера (с указанием авторов документов), кнопки навигации по списку (в случае доступности нескольких листов списка), в зависимости от статуса документа доступны различные действия с документами (для отправленных на проверку документов возможно редактирование, подтверждение и возврат пользователю, для остальных документов - просмотр)

24.1

Сценарий: Создание документа: пользователь

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу создания документа, ввод названия документа не соответствующего стандарту

Тип теста: негативный

Ожидаемый результат: Вывод на странице сообщения о неверном формате названия

24.2

Сценарий: Создание документа: пользователь

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу создания документа, ввод названия документа, соответствующего стандарту, некорректное заполнение полей формы документа, нажатие кнопки «Сохранить»

Тип теста: общий

Ожидаемый результат: Сохранение данных документа на сервисе, переход на страницу со списком документов (список содержит название нового файла, документ доступен для редактирования)

24.3

Сценарий: Создание документа: пользователь

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу создания документа, ввод названия документа, соответствующего стандарту, некорректное заполнение полей формы документа, нажатие кнопки «Завершить», подтверждение желани

завершить документ во всплывшем окне, содержащем количество ошибок

Тип теста: общий

Ожидаемый результат: Сохранение данных документа на сервисе, переход на страницу со списком документов (список содержит название нового файла, документ доступен для просмотра)

24.4

Сценарий: Создание документа: пользователь

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу создания документа, ввод названия документа, соответствующего стандарту, некорректное заполнение полей формы документа, нажатие кнопки «Завершить», отмена завершения документа во всплывшем окне, содержащем количество ошибок

Тип теста: негативный

Ожидаемый результат: Неверно заполненные поля формы документа выделены красным цветом.

24.5

Сценарий: Создание документа: менеджер

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу создания документа, ввод названия документа, соответствующего стандарту, корректное заполнение полей формы документа, нажатие кнопки «Завершить»

Тип теста: негативный

Ожидаемый результат: Вывод сообщения об ошибке: невозможно сохранить файл

24.6

Сценарий: Редактирование документа

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу редактирования документа (путем нажатия на соответствующую ссылку в списке документов)

Тип теста: общий

Ожидаемый результат: Открытие страницы редактирования документа, поля формы содержат сохраненные ранее поля документа и доступны для редактирования, название файла отображается, но недоступно для редактирования.

24.7

Сценарий: Редактирование документа

Тестируемая функциональность: Работа с документом

Описание: Переход на страницу редактирования документа (путем изменения ссылки на страницу: указание заведомо ложного идентификатора документа)

Тип теста: негативный

Ожидаемый результат: Открытие страницы редактирования документа, поля формы не содержат данных

25.1

Сценарий: Изменение статуса документа: менеджер

Тестируемая функциональность: Изменение статуса документа

Описание: Завершение редактирования документа путем нажатия на соответствующую ссылку в списке документов

Тип теста: общий

Ожидаемый результат: Обновление страницы со списком документов, статус документа изменен

25.2

Сценарий: Изменение статуса документа: менеджер

Тестируемая функциональность: Изменение статуса документа

Описание: Возвращение документа на редактирование пользователю путем нажатия на соответствующую ссылку в списке документов

Тип теста: общий

Ожидаемый результат: Обновление страницы со списком документов, статус документа изменен

3.4 Тестирование безопасности

26.1 – Шифрование данных при хранении

Сценарий: Получение пользовательских данных из cookie

Описание: Размещение вредоносного скрипта, выводящего пользовательские cookie: `<script>alert (document.cookie) ;</script>`

Тип теста: Cross-site scripting

Ожидаемый результат: Cookie зашифрованы

26.2 – Шифрование данных при передаче

Сценарий: Перехват пользовательских данных

Описание: Анализ сетевого трафика пользователя системы

Тип теста: Sniffer

Ожидаемый результат: Перехваченные данные зашифрованы

27 – Доступ к запрещенным функциям

Сценарий: Получение доступа к функциональности

Описание: Получение доступа к функциям системы пользователями разных ролей путем открытия соответствующих страниц с подменой параметров открытия документа в адресной строке (status, fid).

Тип теста: Authorization Bypass

№	Роль	Функциональность	Ожидаемый результат
27.1	Не аутентифицирован	Создание документа	Ошибка
27.2	Пользователь	Создание документа	Документ создан
27.3	Менеджер	Создание документа	Ошибка
27.4	Не аутентифицирован	Просмотр документа	Ошибка
27.5	Пользователь: автор документа	Просмотр документа	Просмотр разрешен
27.6	Менеджер: закреплен за автором документа	Просмотр документа	Просмотр разрешен
27.7	Пользователь: не автор документа	Просмотр документа	Ошибка
27.8	Менеджер: не закреплен за автором документа	Просмотр документа	Ошибка
27.9	Не аутентифицирован	Изменение статуса «Формируется» на статус «Отправлен на проверку»	Ошибка
27.10	Пользователь: автор документа	Изменение статуса «Формируется» на статус «Отправлен на проверку»	Статус изменен
27.11	Менеджер: закреплен за автором документа	Изменение статуса «Формируется» на статус «Отправлен на проверку»	Ошибка
27.12	Пользователь: не автор документа	Изменение статуса «Формируется» на статус «Отправлен на проверку»	Ошибка
27.13	Менеджер: не закреплен за автором документа	Изменение статуса «Формируется» на статус «Отправлен на проверку»	Ошибка
27.14	Не аутентифицирован	Редактирование документа со статусом «Формируется»	Ошибка
27.15	Пользователь: автор документа	Редактирование документа со статусом «Формируется»	Результаты редактирования сохранены
27.16	Менеджер: закреплен за автором документа	Редактирование документа со статусом «Формируется»	Ошибка
27.17	Пользователь: не автор документа	Редактирование документа со статусом «Формируется»	Ошибка
27.18	Менеджер: не	Редактирование	Ошибка

	закреплен за автором документа	документа со статусом «Формируется»	
27.19	Не аутентифицирован	Редактирование документа со статусом «Отправлен на проверку»	Ошибка
27.20	Пользователь: автор документа	Редактирование документа со статусом «Отправлен на проверку»	Ошибка
27.21	Менеджер: закреплен за автором документа	Редактирование документа со статусом «Отправлен на проверку»	Результаты редактирования сохранены
27.22	Пользователь: не автор документа	Редактирование документа со статусом «Отправлен на проверку»	Ошибка
27.23	Менеджер: не закреплен за автором документа	Редактирование документа со статусом «Отправлен на проверку»	Ошибка
27.24	Не аутентифицирован	Изменение статуса «Отправлен на проверку» на статус «Подтвержден»	Ошибка
27.25	Пользователь: автор документа	Изменение статуса «Отправлен на проверку» на статус «Подтвержден»	Ошибка
27.26	Менеджер: закреплен за автором документа	Изменение статуса «Отправлен на проверку» на статус «Подтвержден»	Статус изменен
27.27	Пользователь: не автор документа	Изменение статуса «Отправлен на проверку» на статус «Подтвержден»	Ошибка
27.28	Менеджер: не закреплен за автором документа	Изменение статуса «Отправлен на проверку» на статус «Подтвержден»	Ошибка
27.29	Не аутентифицирован	Редактирование документа со статусом «Подтвержден»	Ошибка
27.30	Пользователь: автор документа	Редактирование документа со статусом «Подтвержден»	Ошибка
27.31	Менеджер: закреплен за автором документа	Редактирование документа со статусом «Подтвержден»	Ошибка
27.32	Пользователь: не автор документа	Редактирование документа со статусом «Подтвержден»	Ошибка

27.33	Менеджер: не закреплен за автором документа	Редактирование документа со статусом «Подтвержден»	Ошибка
27.34	Не аутентифицирован	Изменение статуса «Формируется» на статус «Подтвержден»	Ошибка
27.35	Пользователь: автор документа	Изменение статуса «Формируется» на статус «Подтвержден»	Ошибка
27.36	Менеджер: закреплен за автором документа	Изменение статуса «Формируется» на статус «Подтвержден»	Ошибка
27.37	Пользователь: не автор документа	Изменение статуса «Формируется» на статус «Подтвержден»	Ошибка
27.38	Менеджер: не закреплен за автором документа	Изменение статуса «Формируется» на статус «Подтвержден»	Ошибка

28 – Проверка вводимых пользователем данных

Сценарий: Вставки исполняемого кода

Описание: Ввод в поле пароль: `testpass' OR '1'='1`

Тип теста: Code injections

Ожидаемый результат: Ошибка

29.1 – Подмена данных на стороне клиента

Сценарий: Подмена идентификатора сессии

Описание: Установление в cookie ticket идентификатора, дублирующего ранее выданный

Тип теста: Authorization Bypass

Ожидаемый результат: Ошибка

29.2 – Подмена данных на стороне клиента

Сценарий: Подмена роли пользователя

Описание: Установление в cookie role роли, не соответствующей действительной роли пользователя

Тип теста: Authorization Bypass

Ожидаемый результат: Ошибка

3.5 Покрывтие кода тестами

Цикломатическая сложность кода: 97 (посчитано автоматически специализированным средством jsmeter), число модульных тестов: 76, итого покрытие кода тестами составляет $76/97 * 100\% = 76,3\%$

3.6 Пример реализации тестов

Тестирование проводилось с помощью библиотеки QUnit, позволяющей писать тесты для кода на javascript.

Пример блочного теста:

```
QUnit.test( 'ticketvalidation()', function() {
    setCookie('ticket','c2e0d234eb89219e6ecee7fb9a747a4a')
    ok(ticketvalidation(), "Нормальный идентификатор сессии" );

    delCookie('ticket');
    ok(!ticketvalidation(), "Cookie не содержит `ticket`" );

    setCookie('ticket',null)
    ok(!ticketvalidation(), "Cookie содержит `ticket`= null" );

    setCookie('ticket','')
    ok(!ticketvalidation(), "Cookie содержит `ticket`= "" );

    setCookie('ticket','c2e0d234eb89219e6ecee7fb9a747a4')
    ok(!ticketvalidation(), "Cookie содержит `ticket`длины 31 байт" );

    setCookie('ticket','c2e0d234eb89219e6ecee7fb9a747a4aa')
    ok(!ticketvalidation(), "Cookie содержит `ticket`длины 31 байт" );
});
```

Пример интеграционного теста:

```
QUnit.test( 'numpage()', function() {
    setCookie('ticket','6176afa8c488b2367e04321c9a031578');
    var pages;
    $.ajax({
        type: "POST",
        async: false,
        url:CARDSLIST,
        data:{'ticket':getCookie(TICKET)},
        dataType:"text"
    })
    .done( function(data){
        pages = Math.ceil($.parseJSON(data)[0].count/FILEINPAGE);
    })
});
```



```

equal(numpage(), pages, "Действительный идентификатор сессии" );

CARDSLIST = "http://some.bad.url";
equal(numpage(), 50, "Действительный идентификатор сессии, неверный
ответ сервиса");

delCookie('ticket');
equal(numpage(), -1, "Идентификатор сессии не установлен" );
});

```

4. Результаты тестирования

4.1 Сводная таблица по тестированию

Номера тестов	Сценарий	Найденные ошибки	Дата проведения
Блочное тестирование			
1.1-1.6	ticketvalidation()	0	26.12.2014
2.1-2.12	logininvalidation()	0	26.12.2014
3.1-3.4	login(login, pswd)	0	26.12.2014
4.1-4.2	logout()	Ошибка 1	26.12.2014
5.1-5.2	headerload()	0	26.12.2014
5a.1-5a.2	loginmessage()	0	26.12.2014
6.1-6.2	securityload()	0	26.12.2014
7.1-7.3	numpage()	0	26.12.2014
8.1-8.6	statuschange(fid,newstatus)	Ошибка 2	26.12.2014
9.1-9.7	cardstemplate(role,documentinfo)	0	26.12.2014
10.1-10.8	tablefooter(page, maxpage)	0	26.12.2014
11.1-11.5	cardsload()	0	26.12.2014
12.1-12.3	cardformvalidation()	Ошибка 3	26.12.2014
13.1-13.4	filenamevalidation(filename)	Ошибка 3	26.12.2014
14.1-14.5	cardload()	0	26.12.2014
15.1-15.4	cardsave()	0	26.12.2014
Интеграционное тестирование			
16.1-16.2	headerload(),ticketvalidation()	0	26.12.2014
17.1-17.3	loginmessage(), ticketvalidation()	0	26.12.2014
18.1-18.2	securityload(), ticketvalidation()	0	26.12.2014
19.1-19.3	numpage (),ticketvalidation()	0	26.12.2014
20.1-20.5	cardsload(),numpage(), cardstemplate(), tablefooter()	0	26.12.2014

21.1-21.5	cardsave(),filenamevalidation(), cardformvalidation()	0	26.12.2014
Тестирование безопасности			
26.1-26.2	Шифрование	Ошибка 4	26.01.2015
27.1-27.38	Доступ к функциям	0	26.01.2015
28	Проверка вводимых данных	0	26.01.2015
29.1-29.2	Подмена данных	Ошибка 5	

Все аттестационные тесты (22.1 – 25.2) были пройдены успешно.

4.2 Примеры найденных ошибок

Ошибка 1:Тест 4.1: удаление данных о сессии после выхода из системы.

Ожидается, что все данные о пользователе будут удалены, однако роль пользователя не удаляется из cookie. Т.к. роль пользователя не может быть использована без идентификатора сессии ошибка является несущественной.

Вариант исправления: добавить вызов delCookie('role')

Ошибка 2:Тест 8.6: изменение статуса документа пользователем.

Ожидается, что запрос к сервису не будет осуществляться в связи с недостатком прав, однако запрос осуществляется. Т.к. проверка на роль пользователя дополнительно осуществляется на сервисе, данная ошибка не может быть использована как уязвимость и не является критической.

Вариант исправления: добавить проверку на роль авторизованного пользователя.

Ошибка 3:Тест 12.1, 13.3: ввод символов кириллицы и латинского алфавита.

Ожидается, что ввод любых символов кириллицы не приведет к возникновению ошибки, однако буквы Ё и ё обрабатываются как запрещенные символы. В связи с нечастым использованием этих букв в текстах документов ошибка не является критической.

Вариант исправления: внести Ё и ё в регулярные выражения.

Ошибка 4:Тест 26.1, 26.2: шифрование при хранении и передаче данных.

Ожидается, что хранимые и передаваемые данные находятся в зашифрованном виде, однако на практике все данные находятся в открытом виде.

Вариант исправления: добавить шифрование данных.

Ошибка 5:Тест 29.1: Подмена идентификатора сессии

Ожидается, что подмена идентификатора сессии приведет к возникновению ошибки, однако подмена на действительный идентификатор другого пользователя приводит к предоставлению злоумышленнику прав этого пользователя.