

Отчёт о тестировании проекта Visionary






SCRUM-113 «Получение, обновление, удаление изображений»

Цель

фичи:

Реализовать CRUD-эндпоинты для изображений с поддержкой листинга, пагинации и фильтрации по protection.

1. GET single image by UUID



Сценарий	Запрос	Ожидание	Факт	Статус
Public-image	GET /api/v1/image/{publicUuid}	200 + ImageResponse	200	
Private-image владельцем	GET /api/v1/image/{privateUuid} + JWT (owner)	200 + ImageResponse	200	
Private-image чужим	GET /api/v1/image/{privateUuid} + JWT (other)	403 ACCESS_FORBIDDEN	403	
Несуществующий UUID	GET /api/v1/image/{randomUuid}	404 IMAGE_NOT_FOUND	404	
Без токена (private)	GET /api/v1/image/{privateUuid} без JWT	401 Unauthorized	401	

2. PUT update protection



Сценарий	Запрос	Ожидание	Факт	Статус
Владелец меняет protection	PUT /api/v1/image/{uui d} + JWT(owner)	200 + обновлённый ImageResponse	200	✓
Чужой пытается	PUT /api/v1/image/{uui d} + JWT(other)	403 ACCESS_FORBIDDEN	403	✓
Без токена	PUT /api/v1/image/{uui d} без JWT	401 Unauthorized	401	✓

3. DELETE image

Сцена рий	Запро с	Ожидани е	Факт	Стат ус
Владе лец удаляет	DELET E /api/v1/imag e/{uuid} + JWT(owner)	200; затем GET → 404 IMAGE_NOT_F OUND	200→40 4	✓
Админ удаляет чужое	DELET E /api/v1/imag	200	200	✓



	e/{uuid} + JWT(admin)			
Чужой без прав	DELETE E /api/v1/image/{uuid} + JWT(other)	403 ACCESS_DENIED	403	
Без токена	DELETE E /api/v1/image/{uuid} без JWT	401 Unauthorized	401	

4. GET image list (current user)

Сценари й	Запрос	Ожидани е	Факт	Статус
Валидны й: size/p age + protec tion + JWT	GET /api/v1 /user/i mages ?size=. ..&prot ection =... + JWT	200 + масси в изобр ажени й	200	
invalid protec tion	...?protec tion=in valid + JWT	400 INVALI D_REQ UEST	400	
Пользова тель удалё н	для удалё нного ID + JWT	410 USER_ DELET ED	410	
Без токена	GET ... без JWT	401 Unaut horize d	401	

5. GET image list (other user)

Сценари й	Запрос	Ожидан ие	Факт	Статус
Активны й чужой + JWT	GET /api/v1/user/{o	200 + только public	200	





	therId}/images ?... + JWT			
Неактив ный чужой	для удалённого ID + JWT	410 USER_DELETE D	410	
Без токена	без JWT	401 Unauthorized	401	

SCRUM-98 «Регистрация и авторизация»

Цель

Фичи:

Полный end-to-end поток регистрации, логина и ротации JWT (access + refresh).

Сцена рий	Запро с	Ожид ание	Факт	Статус
Регист рация нового пользовате ля	POST /api/v1/auth/ register + валидный	200 + AuthRespon se (tokens)	200	
Регист рация: дубликат email	POST register + существую щий email	409 EMAIL_ALRE ADY_IN_USE	409	
Регист рация: дубликат nickname	POST register + существую щий nickname	409 NICKNAME_ ALREADY_IN _USE	409	
Логин корректный	POST /api/v1/auth/ login + валидные creds	200 + AuthRespon se	200	

Логин неверный пароль	POST login + wrong password	401 INVALID_CREDENTIALS	401	✓
Refresh в валидного токена	POST /api/v1/auth/ refresh + valid token	200 + новые токены	200	✓
Refresh просроченн ого токена	POST refresh + expired token	401 EXPIRED_REFRESH_TOKEN	401	✓
Refresh несуществу ющего токена	POST refresh + random token	401 INVALID_REFRESH_TOKEN	401	✓

SCRUM-112 «Сохранение изображений на сервере»

Цель

Фичи:

POST /api/v1/image/save — загрузка multipart, сохранение в MinIO и БД.

Сценарий	Запрос	Ожидание	Факт	Статус
Валидный JPEG	POST save + JPEG < maxSize + JWT	200 + SaveImageResponse	200	✓

Пустой файл	POST save + файл size=0	400 EMPTY_FILE	400	✓
Неподдер ж. формат	POST save + .png или .txt	415 FILE_FORMAT_ NOT_SUPPORT ED	415	✓
Файл > maxSize	POST save + >100 MB	413 FILE_TOO_LARG E	413	✓
Без токена	POST save без JWT	401 Unauthorized	401	✓
Без file part	POST save без части file	400 INVALID_REQU EST	400	✓

SCRUM-111 «Обновление / Удаление профиля пользователя»


Цель

фичи:

PUT /api/v1/user/{id}, DELETE /api/v1/user/{id}, DELETE /api/v1/user/{id}/avatar.

Сцена рий	Запро с	Ожида ние	Факт	Статус
Updat e own profile	PUT /api/v1/user/ {id} + валидные	200 OK	200	✓

	данные + JWT			
Update чужого профиля	PUT /api/v1/user/ {otherId} + JWT	403 Forbidden	403	✓
Update неактивного пользователя	PUT /api/v1/user/ {deletedId} + JWT	401 USER_DELETED	401	✓
Delete own profile	DELETE /api/v1/user/ {id} + JWT	200 OK; при повторном GET → 401	200→401	✓
Delete чужого профиля	DELETE /api/v1/user/ {otherId} + JWT	403 Forbidden	403	✓
Delete own avatar	DELETE /api/v1/user/ {id}/avatar + JWT	200 OK; avatarUuid → null	200	✓





Delete чужого avatar	DELETE /api/v1/user/{otherId}/avatar + JWT	403 Forbidden	403	
----------------------	--	---------------	-----	---

SCRUM-105 «Получение информации о пользователе»

Цель

фичи:

GET /api/v1/user/{userId}

Сценарий	Запрос	Ожидание	Факт	Статус
Успешное получение	GET /api/v1/user/1 + JWT	200 OK + UserResponse	200	
Пользователь не найден	GET /api/v1/user/999 + JWT	404 USER_NOT_FOUND	404	
Пользователь удалён	GET /api/v1/user/{deletedId} + JWT	410 USER_DELETED	410	
Без токена	GET /api/v1/user/1 без JWT	401 Unauthorized	401	

SCRUM-103 «Получение изображений пользователя»

Цель

фичи:

Эндпоинты для просмотра own/other user images с фильтром по protection и пагинацией.

Тесты дублируют пункты из SCRUM-113 (листинг own и чужих изображений) и GET single image.

SCRUM-102 «Обработка изображений через нейросеть»

Цель фичи:





Предоставить список стилей.

Асинхронная обработка задания через RabbitMQ и Megamind.

Надёжное сохранение задач и результатов.

Сцена рий	Запро с	Ожида ние	Факт	Статус
GET /styles + JWT	GET /api/v1/style s + JWT	200 + StylesRespo nse	200	✓
GET /styles без JWT	GET /api/v1/style s без JWT	401 UNAUTHORI ZED	401	✓
POST /process успешный	POST /api/v1/proc ess?style=...	200 + ProcessingI mageRespo nse { uuid }	200	✓

	&strength=... + JPG + JWT			
POST /process без JWT	без Authorization	401 UNAUTHORI ZED	401	✓
POST /process unknown style	style=f oo + JPG + JWT	404 STYLE_NOT_ FOUND	404	✓
POST /process пустой файл	Multip artFile size=0 + JWT	400 EMPTY_FILE	400	✓
POST /process неподдерж. формат	.pdf или .txt + JWT	415 FILE_FORMA T_NOT_SUP PORTED	415	✓
POST /process strength вне [0,1]	strengt h=-0.1 или 1.1 + JPG + JWT	400 INVALID_RE QUEST	400	✓
Mega mind → 500	Mega mindClient throws HTTP 500	502 MEGAMIND_ ERROR	502	✓

cron-job updateStyles	ручной вызов updateStyles()	Репозиторий обновлён	—	
Listener: timeout	message threshold	Статус → CANCELED	—	
Listener: normal processing	Mega mind OK	PROCESSING → COMPLETED	—	
Listener: Megamind error	Mega mindClient throws ServiceException	Статус → FAILED	—	

SCRUM-121 «Splash-screen»

Цель

Фичи:

При старте показывать Splash-screen с логотипом, затем авто-переход на LoginScreen.

Сцена рий	Действие	Ожидание	Факт	Статус
--------------	----------	----------	------	--------

Появление Splash-screen	Холодный старт приложения	Логотип на белом фоне	✓	✓
Длительность ≈1,25 с	Измерить время	≈1,25 сек	1,27 сек	✓
Авто-переход на LoginScreen	По таймеру	Переход на экран входа	✓	✓
Back-кнопка на Splash	Нажать «Back»	Закрытие/сворачивание приложения	✓	✓
Поворот экрана во время Splash	Повернуть устройство	Логотип остаётся центрированным, таймер идёт	✓	✓

SCRUM-86 «Экран настроек»

Цель

фичи:

Экран «Настройки» с набором пунктов (Subscription, Pro/Premium, Notify, Account, Change Password, Logout, Delete Account, Personalization, Change Theme).

Компонент: SettingsItem + mock-диалоги.

Пункт меню	Ожидаемое поведение	Факт	Статус
Подписка	Открытие proDialog	✓	✓
Premium-версия	Открытие premiumDialog	✓	✓
Уведомления	Переключение тумблера	✓	✓
Аккаунт	Без действий (mock)	✓	✓
Сменить пароль	Появление поля ввода и скрывание по check	✓	✓
Выйти из аккаунта	Mock-диалог подтверждения	✓	✓
Удалить аккаунт	Mock-диалог удаления	✓	✓

Персонализация	Открытие mock-диалога		
Сменить тему	Переключение UI-цвета (mock)		
Скролл списка	Корректная прокрутка		
Ландшафтная ориентация	Адаптивность, кликабельность без обрезки текста		

SCRUM-82 «Экраны профиля»

Фича:

MyProfileScreen и OtherUserProfileScreen (заглушки).

Элемент	Действие	Ожидание	Факт	Статус
Аватар (свой/чужой)	Тап	Без падений		
Поле описания	Тап	Фокус, ввод запрещён		

Иконки статистики	Тап	Без ошибок	✓	✓
Кнопки «Публикации/Личное»	Тап	Переключатели	✓	✓
Элемент грид-изображения	Тап	Без падений	✓	✓

SCRUM-123 «Камера и галерея»

Фича:

Подключить выбор/съёмку изображения (CreatePictureScreen).

Элемент	Действие	Ожидание	Факт	Статус
«Сделать фото»	Тап	Запрос разрешений → открытие камеры	✓	✓
«Выбрать из галереи»	Тап	Запрос разрешений	✓	✓



		→ открытие галереи		
Permissions CAMERA / STORAGE	Согласие/отказ	Корректная обработка разрешений	✓	✓
Полученный URI	После операции	URI сохраняется в состоянии компонента	✓	✓
Отображение фото	На экране	Показ выбранного изображения	✓	✓

SCRUM-117 «Настроить метрики»

Фича:

Интеграция Spring Actuator + Prometheus.

Endpoint	Метод	Ожидание	Факт	Статус
/actuator/prometheus	GET	200 OK + Prometheus формат		
/actuator/health	GET	200 OK + статус UP		

/actuat or/metrics/h ttp.server.re quests	GET	200 OK + HTTP метрики		
/v3/api -docs	GET	200 OK + JSON OpenAPI	