Методы и модели в экономике

Автоматизированное выявление паттерна фирменного стиля предприятия

А.В. Фирсов, П.Е. Каршаков, А.В. Шелкова

Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство)», г. Москва, Россия

Аннотация. В статье рассматривается задача автоматизированного выявления и формализации паттернов фирменного стиля веб-ресурсов промышленных предприятий, функционирующих в составе автоматизированных систем управления. Предложен алгоритмический подход к анализу визуальных и структурных характеристик, позволяющий выделять устойчивые закономерности оформления и представления информации. Разработана методика описания паттернов в виде формализованных элементов, обладающих атрибутами и связями, что обеспечивает возможность их последующего использования при формировании согласованных конфигураций. Показано, что автоматизация процесса выявления паттернов позволяет повысить унификацию интерфейсных решений, обеспечить их совместимость и упростить интеграцию с информационными системами.

Ключевые слова: *брендбук, веб-ресурс, автоматизированная система управления, информационная система, визуальная структура, цифровая инфраструктура, Figma, плагин.*

DOI: 10.14357/20790279260307 **EDN:** UNUEDM

Введение

В современных исследованиях отмечается, что веб-ресурсы становятся неотъемлемой частью цифровой инфраструктуры предприятий, обеспечивая интеграцию информационных потоков и доступ к корпоративным данным [1]. Веб-интерфейсы перестают выполнять исключительно презентационную функцию и становятся частью производственно-информационных процессов, обеспечивая доступ к технической документации, каталогам продукции, сервисным функциям и данным корпоративных систем. В этой связи требования к согласованности, структурированности и унификации представления информации приобретают принципиальное значение.

Фирменный стиль рассматривается как комплекс визуальных и коммуникационных средств, обеспечивающих целостность восприятия бренда и его идентификацию в информационном пространстве [1]. Таким образом, элементы фирменного стиля могут быть интерпретированы как структурные компоненты, влияющие на целостность информационной среды и подлежащие стандартизации.

Рост сложности корпоративных информационных систем требует применения унифицированных подходов к представлению данных и интерфейсных решений [4]. В таких условиях отсутствие формализованного подхода к описанию и применению элементов фирменного стиля приводит к фрагментации интерфейсных решений, увеличению трудозатрат на разработку и сопровождение, а также снижению эффективности взаимодействия пользователей с системой.

Традиционные методы формирования фирменного стиля основываются на разработке регламентирующих документов, известных как брендбуки или гайдлайны. При этом традиционные брендбуки не обеспечивают достаточной степени формализации, что ограничивает их применение в автоматизированных системах разработки [4]. Однако данные документы, как правило, создаются вручную и не обладают достаточной степенью формализации, что затрудняет их использование в автоматизированных процессах разработки [5]. В связи с этим возникает необходимость перехода к более строгому представлению — паспорту фирменного стиля, который может быть интерпретирован как структурированная модель данных, пригодная для машинной обработки.

Актуальной задачей в данном контексте является разработка методики автоматизированного выявления паттернов фирменного стиля, позволяющей формализовать повторяющиеся элементы и использовать их при проектировании и модернизации веб-ресурсов. Решение данной задачи направлено на повышение согласованности интерфейсов, улучшение интеграции с системами и повышение эффективности разработки программного обеспечения.

1. Анализ существующих подходов

Традиционный процесс разработки брендбука включает ручное формирование визуальных и текстовых элементов, что делает его трудоемким и зависимым от субъективных факторов [1]. Данный процесс включает формирование цветовой палитры, описание типографики, фиксацию правил использования логотипа и других элементов визуальной идентификации. Несмотря на широкое распространение, такой подход характеризуется высокой трудоемкостью и низкой степенью формализации.

Ручная обработка данных в процессе проектирования часто приводит к ошибкам, связанным с некорректной передачей параметров и человеческим фактором [1].

Дополнительной проблемой является сложность актуализации документа. В условиях итеративной разработки изменения в фирменном стиле происходят достаточно часто, и каждый раз требуется обновление паспорта. При ручном подходе это приводит к дублированию работы и снижению эффективности процессов.

Современные инструменты проектирования интерфейсов предоставляют возможности централизованного хранения стилевых параметров. В частности, системы наподобие Figma позволяют задавать цветовые и текстовые стили, которые используются в рамках проекта. При этом можно использовать облачные технологии, что является актуальной задачей при разработке веб-ориентированных информационных систем [6]. Данный специализированный редактор позволяет совместную удаленную работу над UI/UX-проектами [7]. Figma поддерживает интеграцию с системами управления проектами (Jira, Asana), инструментами для генерации кода (Zeplin) и другими сервисами, что упрощает рабочий процесс [8]. Однако встроенные средства автоматизированного формирования структурированного документа на основе этих данных отсутствуют. Существующие программные решения частично решают данную задачу, но либо ограничены функционально, либо требуют значительных усилий по настройке и адаптации. Несмотря на развитие инструментов проектирования, автоматизация формирования документации на основе дизайн-систем остается ограниченной [9].

Таким образом, анализ существующих подходов показывает, что на текущий момент отсутствуют универсальные методы, обеспечивающие автоматизированное выявление паттернов фирменного стиля и их формализацию в виде структурированного паспорта. Это обуславливает необходимость разработки соответствующей методики.

2. Методика автоматизированного выявления паттернов

В основе предлагаемой методики лежит представление фирменного стиля как совокупности паттернов — устойчивых повторяющихся элементов, характеризующих визуальную и структурную организацию интерфейса. Паттерны могут быть выделены на основе анализа стилевых параметров, используемых в проектной среде. Использование паттернов как повторяющихся элементов широко применяется в современных системах проектирования и анализа интерфейсов [9].

Ключевым этапом является формализация данных, описывающих элементы фирменного стиля. Цветовые характеристики представляются в виде числовых значений (например, RGB или HEX), текстовые параметры — в виде набора атрибутов, включающих семейство шрифта, размер, интерлиньяж и другие характеристики. Представление элементов фирменного стиля в виде структурированных данных позволяет использовать их в автоматизированных системах обработки информации [10].

Процесс выявления паттернов включает извлечение данных из проектной среды, их классификацию и структурирование. На данном этапе осуществляется анализ локальных стилей, определяются их типы и взаимосвязи. Особое внимание уделяется фильтрации некорректных или неполных данных, что позволяет повысить точность последующей генерации документа.

Результатом данного процесса является формализованная модель фирменного стиля, которая может быть использована для автоматизированного формирования паспорта. Таким образом, осуществляется переход от неструктурированного набора параметров к системному представлению, пригодному для интеграции с программными средствами. Методы структурирования и классификации данных позволяют повысить точность и надежность автоматизированных решений [4].

Современная среда Figma предоставляет API для создания плагинов, которые могут автоматически читать стили документа и генерировать новые элементы интерфейса и документации. Использование API проектных систем позволяет автоматизировать извлечение и обработку данных в рамках единой среды разработки [10]. Это делает ее удобной платформой для реализации автоматизированных систем анализа и генерации дизайн-систем.

Существующие решения в виде готовых плагинов обладают ограниченной функциональностью (табл. 1).

Анализ показывает, что существующие плагины либо являются платными, либо формируют упрощенный результат, не учитывающий индивидуальные особенности дизайн-системы. Поэтому разработка собственного решения, обладающего гибкостью, бесплатностью и возможностью настройки, является актуальной задачей.

На основе анализа существующих решений были сформулированы функциональные и нефункциональные требования к разрабатываемому плагину (рис. 1).



Рис. 1. Функциональные и нефункциональные требования к плагину

3. Реализация программного инструмента

Для апробации предложенной методики был разработан программный инструмент в виде плагина, интегрированного в среду проектирования интерфейсов Figma. Разработка плагина являлась непосредственной реализацией предложенного подхода автоматизированного выявления паттернов, поскольку именно через него осуществляется извлечение, структурирование и преобразование стилевых данных в формализованное представление фирменного стиля.

Процесс разработки начался с проектирования архитектуры системы, основанной на разделении логики обработки данных и пользовательского интерфейса. Такая декомпозиция позволила отделить функциональную часть, отвечающую за анализ и генерацию элементов брендбука, от слоя взаимодействия с пользователем, что обеспечило гибкость и расширяемость решения.

В рамках архитектуры плагин включает три ключевых компонента: манифест (manifest.json), основной код (code.js) и пользовательский интерфейс (ui.html). Манифест определяет конфигурацию плагина, включая доступ к текущему документу и точки входа в систему. Основной поток отвечает за реализацию алгоритмов извлечения стилевых данных, а интерфейс обеспечивает управление процессом генерации и передачу параметров в основную логику.

Схема взаимодействия компонентов реализована следующим образом: пользователь запускает плагин и инициирует интерфейсную часть, после чего задает параметры генерации (например, выбор разделов цветовой палитры и типографики). Далее UI передает сообщение в основной поток, где осуществляется обращение к API Figma и извлечение локальных стилей через методы figma.

Табл. 1

Плагины для генерации гайдлайнов

Название плагина	Возможности	Ограничения
Automatic Style Guides	Создает страницу с цветами и текстовыми стилями	Ограниченный дизайн, нет кастомизации
Brand Manual Generator	Работает с переменными Figma	Требует платной подписки
AI Brand Guide	Использует ИИ для описания цветов	Сложность настройки

Полученные данные проходят этап обработки и структурирования, после чего формируется новый документ или страница, содержащая сгенерированный паспорт фирменного стиля. Архитектура в виде блок-схемы представлена на рис. 2.

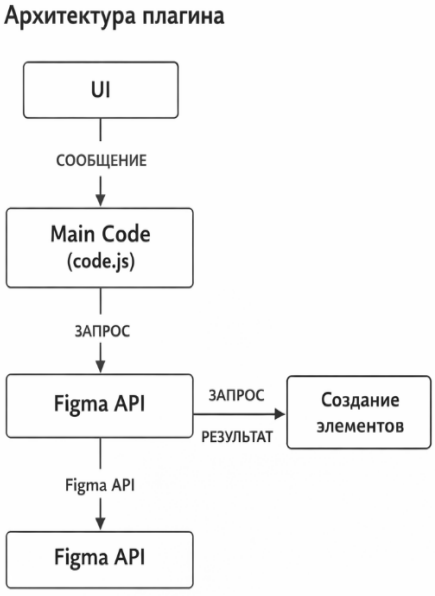


Рис. 2. Блок-схема архитектуры плагина

В процессе реализации ключевым этапом стало проектирование алгоритма buildGuide(), который объединяет все стадии генерации в единый процесс. В рамках его работы выполняется загрузка базовых шрифтов, формирование корневого контейнера с использованием Auto Layout, структурирование элементов и последовательная генерация разделов. Создаваемая структура страницы включает цветовые и текстовые паттерны, представленные в виде карточек с параметрическим описанием.

Пользовательский интерфейс плагина был разработан с учетом принципов минимизации сложности взаимодействия (рис. 3). Он включает элементы выбора разделов, поле ввода названия страницы и кнопку запуска генерации. После активации процесса интерфейс блокируется, что исключает повторный запуск операции до завершения текущего цикла. Такой подход обеспечивает контроль состояния системы и предотвращает конфликтные вызовы.

Особое внимание в процессе реализации было уделено устойчивости работы системы. В коде предусмотрена обработка ошибок, связанных с отсутствием локальных стилей, некорректной структурой данных, отсутствием шрифтов и невозможностью создания страниц. В случае возникновения ошибок система не прерывает выполнение полностью, а формирует корректное уведомление для пользователя и сохраняет целостность документа. Вся логика генерации дополнительно обернута в механизм try...catch, что позволяет обрабатывать непредвиденные сценарии без аварийного завершения работы плагина.

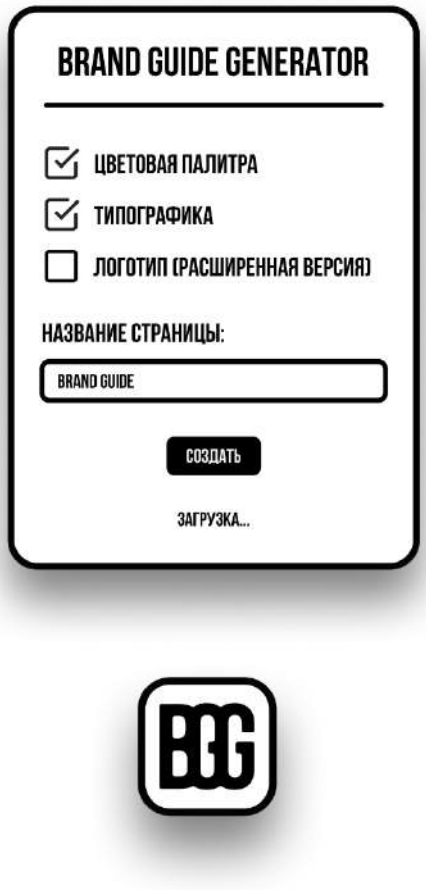


Рис. 3. Реализованное окно плагина в Figma

Таким образом, реализация программного инструмента представляет собой практическое воплощение методики автоматизированного выявления паттернов, в которой процесс извлечения стилевых данных из среды проектирования непосредственно трансформируется в структурированную модель фирменного стиля.

4. Техническая реализация плагина

Для разработки плагина автоматизированного формирования брендбука в среде Figma был выполнен последовательный комплекс работ, включающий настройку среды разработки, созда-

ние проекта, конфигурацию параметров плагина и реализацию программной логики. На начальном этапе была установлена настольная версия Figma Desktop App, поскольку функциональность разработки и тестирования плагинов недоступна в браузерной версии платформы. После установки среды разработчика был создан новый плагин через меню Plugins → Development → New Plugin. При этом был выбран шаблон с пользовательским интерфейсом (with UI), что позволило реализовать взаимодействие между графической частью и основной логикой приложения.

На следующем этапе созданный проект был открыт в среде Visual Studio Code, где сформирована его базовая структура. В состав проекта входят конфигурационный файл manifest.json, основной файл программной логики code.js и файл пользовательского интерфейса ui.html. Такая структура обеспечивает разделение ответственности между компонентами системы, что упрощает сопровождение и дальнейшее расширение функциональности. Структура проекта в среде разработки представлена на рис. 4.

Ключевым элементом конфигурации плагина является файл manifest.json, определяющий параметры его работы в среде Figma. Данный файл задает основные характеристики приложения, включая имя плагина, уникальный идентификатор, версию API и пути к основным модулям системы. В реализованном решении поле name содержит название Brand Guide Generator, а поле id обеспечивает уникальную идентификацию плагина в системе Figma. Поле api определяет используемую версию API, обеспечивающую совместимость с текущей версией платформы.

Параметр main указывает на основной файл логики code.js, в котором реализуются алгоритмы извлечения стилевых данных и генерации брендбука. Параметр ui задает файл интерфейса ui.html, отвечающий за взаимодействие с пользователем. Дополнительно в конфигурации используется параметр documentAccess со значением current-document, предоставляющий доступ к текущему документу Figma, включая локальные цветовые и текстовые стили. Ограничение editorType: [«figma»] определяет среду выполнения плагина, а параметр networkAccess со значением allowedDomains: [«none»] исключает сетевые запросы, обеспечивая полностью локальную обработку данных. Содержимое файла manifest.json представлено на рис. 5.

{} manifest.json > {} networkAccess

```
{
  "name": "Brand Guide Generator",
  "id": "1627671274814146030",
  "api": "1.0.0",
  "main": "code.js",
  "capabilities": [],
  "enableProposedApi": false,
  "documentAccess": "current-document",
  "editorType": ["figma"],
  "ui": "ui.html",
  "networkAccess": {
    "allowedDomains": [
      "none"
    ]
  }
}
```

Рис. 5. Содержимое файла manifest.json

Основная функциональность плагина реализована в файле code.js. При запуске выполняется вызов функции figma.showUI(), которая инициирует отображение пользовательского интерфейса заданного размера. Взаимодействие между интерфейсом и основной логикой осуществляется через механизм сообщений figma.ui.onmessage, позволяющий передавать команды из UI в основной поток обработки.

При получении команды generate-brand-guide запускается процесс генерации брендбука. В зависимости от результата выполнения операции интерфейсу возвращается сообщение о завершении (done) либо об ошибке (error). Такой подход



Рис. 4. Структура проекта в VS Code

обеспечивает асинхронное взаимодействие между компонентами системы и повышает устойчивость работы плагина.

Центральным элементом логики является функция buildGuide(), которая реализует полный цикл формирования брендбука (рис. 6). В начале работы осуществляется загрузка базовых шрифтов Inter Regular и Inter Bold по-средством функции loadBaseFonts(), что обеспечивает корректное отображение текстовых элементов. Далее происходит извлечение локальных стилевых данных документа с использованием API Figma, включая цветовые (figma.getLocalPaintStyles()) и текстовые (figma. getLocalTextStyles()) стили .

После получения данных выполняется проверка наличия целевой страницы. Если она с заданным именем существует, очищается от предыдущих элементов, в противном случае создается новая. Далее формируется корневая структура документа в виде фрейма с вертикальным Auto Layout. Для него задаются параметры отступов между элементами (32 пикселя), внутренние поля (40 пикселей) и нейтральный светло-серый фон, обеспечивающий визуальную универсальность.

Внутри корневого контейнера последовательно формируются основные разделы брендбука. Сначала добавляется заголовок «Brand Guide», а затем краткое описание автоматически сгенерированного гайдлайна. Далее, в зависимости от переданных параметров, формируются разделы цветовой палитры, типографики и блока логотипа. В случае отсутствия соответствующих стилевых данных система формирует информационные сообщения, не прерывая процесс генерации.

После завершения формирования всех элементов корневой фрейм размещается на рабочей странице с координатами (80, 80). Затем автоматически выполняется масштабирование камеры Figma для удобного отображения результата.

Архитектура плагина построена по модульному принципу, что обеспечивается использованием набора вспомогательных функций. Модульный подход к построению программных систем обеспечивает гибкость, масштабируемость и удобство сопровождения программного обеспечения [9]. Функция createTextNode() отвечает за создание текстовых элементов с заданными параметрами оформления. Функция setupVerticalAutoLayout() используется для настройки вертикального расположения компонентов внутри контейнеров. Функция createCardsWrap() формирует контейнер для карточек цветовых стилей с поддержкой переноса строк и горизонтального размещения.

Дополнительно используются утилитарные функции: rgbToHex(), обеспечивающая преобразование цветовых значений в HEX-формат, и isSolidPaintStyle(), выполняющая проверку корректности стилевых данных. Такое разделение логики позволяет повысить читаемость кода и упростить его дальнейшее масштабирование.

Формирование визуальных элементов брендбука реализуется через специализированные функции. Функция createColorCard() создает карточки цветовых стилей, включающие цветовой образец, название и HEX-код, объединенные в единый визуальный компонент с закругленными углами и тонкой обводкой. Функция createFontCard() формирует карточки текстовых стилей, отображающие пример текста с применением реальных параметров шрифта и его технические характеристики.

Дополнительно реализована функция createLogoPlaceholderCard(), предназначенная для создания блока-заглушки логотипа. Данный элемент представлен в виде графического контейнера с рамкой и текстовым плейсхолдером и служит основой для дальнейшего расширения функциональности системы.

Таким образом, разработанный плагин представляет собой модульную программную систему, объединяющую процессы извлечения стилевых данных из Figma, их структурирования и автоматической визуализации в виде брендбука. Реализация данного подхода позволяет существенно повысить

```
    try {
      await buildGuide({
        pageName: msg.pageName,
        includeColors: msg.includeColors,
        includeTypography: msg.includeTypography,
        includeLogoBlock: msg.includeLogoBlock
      });

      figma.ui.postMessage({
        type: 'done',
        message: 'Brand Guide создан'
      });
    } catch (error) {
      console.error(error);
      figma.ui.postMessage({
        type: 'error',
        message: 'Ошибка при генерации'
      });
    }
  };

  async function buildGuide(options) {
    const {
      pageName = 'Brand Guide',
      includeColors = true,
      includeTypography = true,
      includeLogoBlock = false
    } = options;
```

Рис. 6. Функция buildGuide() с полный циклом формирования брендбука

уровень автоматизации дизайнерских процессов, сократить временные затраты на формирование документации и минимизировать вероятность ошибок при ручной обработке стилевых параметров.

5. Экспериментальная апробация

Для оценки эффективности предложенного подхода и реализованного программного инструмента была проведена экспериментальная апробация в реальной среде проектирования Figma. В рамках эксперимента была сформирована модель фирменного стиля, включающая цветовые и текстовые параметры, после чего на ее основе был автоматически сгенерирован паспорт фирменного стиля (рис. 7).

В процессе работы плагина система последовательно извлекала локальные стили, классифицировала их по типу (цветовые и текстовые), после чего преобразовывала в структурированные визуальные элементы. Результатом работы стало формирование страницы брендбука, содержащей карточки цветов и типографических стилей с точным соответствием исходным параметрам.

Полученные результаты подтвердили корректность работы алгоритма: все элементы были корректно извлечены (табл. 2 и 3), а значения параметров полностью соответствовали данным, заданным в Figma. Визуальное представление обеспечило наглядность структуры и удобство дальнейшего использования документа в дизайн-процессе.

Сравнительный анализ с ручным подходом показал значительное сокращение временных затрат. Если традиционное формирование брендбука требует значительного времени на верстку и перенос данных, то автоматизированный процесс занимает считанные секунды. Дополнительно исключается вероятность ошибок, связанных с ручным копированием значений (например, HEX-кодов или параметров шрифтов).

Важным преимуществом также является возможность оперативной актуализации документа. При изменении локальных стилей в Figma достаточно повторно запустить процесс генерации, и брендбук автоматически обновляется в соответствии с текущими параметрами дизайн-системы без необходимости ручного редактирования.



Рис. 7. Сгенерированный плагином брендбук (слева) и брендбук после ручной доработки (справа)

Табл. 2

Цветовые стили

Название стиля	HEX-код	Описание
Основной цвет	#123D1E	Темно-зеленый, базовый цвет айдентики
Дополнительный_1	#37A455	Изумрудный, акцентный цвет
Дополнительный_2	#070605	Черный, цвет для текста и темных акцентов

Табл. 3

Текстовые стили

Название стиля	Параметры	Описание
AG Основной - 100/95	Размер 100px, интерлиньяж 95	Основной текст
Ag Дополнительный_деко - 97.7/Auto	Размер 97.7px, интерлиньяж Auto	Декоративный дополнительный текст
Ag Дополнительный_декора - 97.7/95	Размер 97.7px, интерлиньяж 95	Декоративный дополнительный текст

6. Перспективы развития

Дальнейшее развитие данного направления связано с расширением принципов автоматизированного выявления паттернов и их интеграцией в более сложные информационные системы управления дизайн-процессами. В перспективе данный подход может быть встроен в экосистемы, обеспечивая единые стандарты визуальной идентичности во всех цифровых продуктах.

Применение онтологических моделей открывает возможности для более глубокого семантического анализа и формализации знаний о дизайн-системах. Это создает основу для автоматической генерации интерфейсных решений на основе формализованных знаний о бренде.

Дополнительные возможности связаны с применением генеративных алгоритмов и методов машинного обучения, позволяющих не только фиксировать существующие паттерны, но и предлагать их оптимизацию. В таком случае система сможет анализировать структуру дизайн-системы и предлагать улучшения на основе накопленных данных.

При этом необходимо учитывать ограничения, связанные с качеством исходных данных, требованиями информационной безопасности и необходимостью экспертного контроля. Наиболее эффективным остается гибридный подход, сочетающий автоматизированные методы обработки и профессиональную дизайнерскую валидацию результатов.

Заключение

В статье показана методика автоматизированного выявления паттернов фирменного стиля и их формализации в виде паспорта, пригодного для использования в процессе разработки веб-ресурсов промышленных предприятий. Предложенный подход позволяет перейти от ручных, слабо формализованных методов к системным и масштабируемым решениям.

Реализация программного инструмента подтвердила практическую применимость методики и ее эффективность. Полученные результаты демонстрируют значительное сокращение трудозатрат, повышение точности и улучшение согласованности интерфейсных решений.

Таким образом, автоматизация анализа и формализация фирменного стиля является важным направлением развития программных средств, обеспечивающих поддержку жизненного цикла веб-ресурсов в условиях цифровой трансформации промышленности.

Литература

1. Никитин Д.Л., Пекшеев К.С., Виноградская М.Ю. Анализ общих требований к веб-сайту для производственного предприятия // Дневник науки. 2024. № 8(92). URL: https://dnevniknauki.ru/images/publications/2024/8/technics/Nikitin_Peksheev_Vinogradskaya.pdf (дата обращение: 04.05.2026)
2. Кузнецов Д.Д. Разработка веб-приложения для малого предприятия // Молодой ученый. 2023, № 18 (465). С. 14-16.
3. Муртазина А.Р. Разработка плагина для создания графических элементов на основе методов комбинаторики // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2022. № 1. С. 77-81.
4. Павлидис В.Д., Бражникова А.Н. Современные архитектуры построения web-приложений // Наукосфера. 2025. № 3-2. С. 47-52.
5. Епрынцева Н.А. Веб-решение для автоматизации учета рабочего времени и координации задач на предприятиях // Инженерный вестник Дона. №11 (2025) URL: ivdon.ru/ru/magazine/

archive/n11y2025/10510 (дата обращения: 01.06.2026)
6. Ступина М.В. Использование облачных интернет-технологий в разработке веб-ориентированных информационных систем // Инженерный вестник Дона. №1 (2023) URL: ivdon.ru/ru/magazine/archive/n1y2023/8130 (дата обращения: 01.06.2026)
7. Кононова О.С., Фирсов А.В. Применение методологии Agile при совместной работе над UI/UX-проектами в специализированном редакторе верстки // Инженерный вестник Дона, №1 (2023) URL: ivdon.ru/ru/magazine/archive/n1y2023/8164 (дата обращения: 15.05.2026)
8. Винарчук А.В. Коллаборативный дизайн: как Figma способствует сотрудничеству в проектах // Научный Лидер. 2023. № 32(130). С. 6-7.
9. Figma2Code: Automatic Code Generation Method for Figma Design Drafts / L. Zhu, Y. Feng, H. Zhu [et al.] // Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao. 2025. Vol. 37. No. 2. P. 321-329.
10. Tazkiyah S., Arifin S. Perancangan UI/UX pada Website Laboratorium Energy menggunakan Aplikasi Figma // Jurnal Teknologi Terpadu. 2022. Vol. 8, No. 2. P. 72-78.

Фирсов Андрей Валентинович. Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство)», г. Москва, Россия. Заведующий кафедрой. Доктор технических наук, профессор. Область научных интересов: цифровые методы проектирования в текстильной и легкой промышленности. E-mail: firsov-av@rguk.ru

Каршаков Петр Евгеньевич. Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство)», г. Москва, Россия. Аспирант. Область научных интересов: алгоритмический подходов к выбору и применению паттернов. E-mail:pkarshak@gmail.com (Ответственный за переписку).

Шелкова Анастасия Владимировна. Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство)», г. Москва, Россия. Студент. Область научных интересов: информационные технологии в дизайне. E-mail: shelkovaanastasia2004@gmail.com

Automated Identification of Enterprise Corporate Identity Patterns

A.V. Firsov, P.E. Karshakov, A.V. Shelkova
Kosygin Russian State University (Technologies. Design. Art), Moscow, Russia

Abstract. The article addresses the problem of automated identification and formalization of corporate identity patterns in web resources of industrial enterprises operating within automated management systems. An algorithmic approach to the analysis of visual and structural characteristics is proposed, enabling the identification of stable patterns in interface design and information presentation. A methodology for describing corporate identity patterns as formalized elements with defined attributes and relationships is developed, providing the possibility of their further use in generating consistent interface configurations. It is shown that automation of the pattern identification process improves the standardization of interface solutions, ensures their compatibility, and facilitates integration with enterprise information systems.

Keywords: *corporate identity, brand book, web resource, automated control system, information system, visual structure, digital infrastructure, Figma, plugin.*
DOI: 10.14357/20790279260307 **EDN:** UNUEDM

References

1. Nikitin D.L., Peksheev K.S., Vinogradskaya M.Yu. Analysis of General Requirements for a Manufacturing Enterprise Website. Dnevnik Nauki, 2024, no. 8(92). Available at: https://dnevniknauki.ru/images/publications/2024/8/technics/Nikitin_Peksheev_Vinogradskaya.pdf (accessed: May 4, 2026).
2. Kuznetsov D.D. Development of a Web Application for a Small Enterprise. Molodoy Uchenyy (Young Scientist). 2023;18(465):14–16.
3. Murtazina A.R. Development of a Plugin for Creating Graphic Elements Based on Combinatorial Methods. Modern Science: Actual Problems of Theory and Practice. Series: Natural and Technical Sciences. 2022;1:77–81.
4. Pavlidis V.D., Brazhnikova A.N. Modern Architectures of Web Application Development. Naukosfera. 2025;3-2:47–52.
5. Epryntseva N.A. A Web-Based Solution for Automating Working Time Accounting and Task Coordination at Enterprises. Engineering Journal of Don, 2025, no. 11. Available at: https://ivdon.ru/ru/magazine/archive/n11y2025/10510 (accessed: June 1, 2026).
6. Stupina M.V. Application of Cloud Internet Technologies in the Development of Web-Oriented Information Systems. Engineering Journal of Don, 2023, No. 1. Available at: https://ivdon.ru/ru/magazine/archive/n1y2023/8130 (accessed: June 1, 2026).
7. Kononova O.S., Firsov A.V. Application of the Agile Methodology in Collaborative UI/UX Design Projects Using a Specialized Layout Editor. Engineering Journal of Don, 2023, No. 1. Available at: https://ivdon.ru/ru/magazine/archive/n1y2023/8164 (accessed: May 15, 2026).
8. Vinarchuk A.V. Collaborative Design: How Figma Facilitates Cooperation in Projects. Nauchnyi Lider (Scientific Leader). 2023;32(130):6–7.
9. Zhu L., Feng Y., Zhu H., et al. Figma2Code: Automatic Code Generation Method for Figma Design Drafts. Journal of Computer-Aided Design & Computer Graphics. 2025;37(2):321–329.
10. Tazkiyah S., Arifin S. UI/UX Design for the Energy Laboratory Website Using Figma. Jurnal Teknologi Terpadu. 2022;8(2):72–78.

Firsov Andrey V. Federal State Budgetary Educational Institution of Higher Education Kosygin Russian State University (Technologies. Design. Art), Moscow, Russia. Head of Department, Doctor of Technical Sciences, Professor. Research interests: digital design methods in the textile and light industry. E-mail: firsov-av@rguk.ru

Karshakov Peter E. Federal State Budgetary Educational Institution of Higher Education Kosygin Russian State University (Technologies. Design. Art), Moscow, Russia. Postgraduate Student (PhD Student). Research interests: algorithmic approaches to the selection and application of design patterns. E-mail: pkarshak@gmail.com

Shelkova Anastasia V. Federal State Budgetary Educational Institution of Higher Education Kosygin Russian State University (Technologies. Design. Art), Moscow, Russia. Student. Research interests: information technologies in design. E-mail: shelkovaanastasia2004@gmail.com