

- предоставление возможности пользователю, зарегистрированному в приложении, просматривать работы в режиме просмотра стороннего пользователя, такие как: задание руководителя и задачи исполнителей;
- при авторизации в приложении, пользователю будет доступна работа в соответствии с заранее назначенными на него правами;
- если пользователь является исполнителем, ему должны быть доступны задачи его отдела для взятия в работу, если задача не была уже взята в работу другим пользователем, в таком случае пользователь имеет доступ на просмотр;
- в приложении должна быть реализована система поиска данных по принципу фильтров;
- при сохранении, редактировании данных или выходе закрытии окна на этапе работы, приложение должно выдавать соответствующие сообщения, предупреждения пользователя о не сохранённом процессе;
- приложение должно быть оснащено возможностью администратору изменить владельца задания или задачи, а так же возможностью перезаписать путь к документу, в случае изменения местоположения или удаления файла;
- в системе должно отсутствовать возможность удаления задания и задач, во избежание потери данных и ошибок в связи данных;
- приложение должно быть обеспечено быстрой навигацией по задачам и заданиям;
- должна быть реализована возможность открытия связанных между собой задач и заданий в режиме просмотра с рабочей области пользователя;
- программа должна включать в себя внесения данных цеховых заказов, все необходимые данные и осуществлять быстрый поиск по добавленным данным, с возможностью их редактирования;
- должно быть осуществлена история действий пользователя;
- окна должны быть оснащены кнопкой выхода и возвращения на главное окно программы.

В прототипе системы управления проектами были определены роли, составлены UML-диаграммы прецедентов, описаны главные и альтернативные сценарии использования для каждого прецедента, отображена информационно-логическая модель данных и описана архитектура проекта.

Созданный прототип системы позволяет решить все необходимые задачи, обладает понятным и удобным интерфейсом, который упрощает работу пользователей, и является отличным средством автоматизации управления проектами для предприятия НТЦК ОАО «Гомсельмаш».

К. И. Куш

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **О. М. Дерюжкова**, канд. физ.-мат. наук, доцент

ШАБЛОНЫ GRASP: ЭФФЕКТИВНЫЕ ПРИНЦИПЫ ПРОЕКТИРОВАНИЯ

Шаблоны GRASP (General Responsibility Assignment Software Patterns) представляют собой ключевые принципы и паттерны, помогающие разработчикам эффективно структурировать объектно-ориентированные программы. Они обеспечивают четкое распределение ответственности между классами и объектами, что в итоге улучшает читаемость, расширяемость и поддерживаемость кода. Шаблоны GRASP состоят из 5 основных и 4 дополнительных паттернов (рисунок 1).

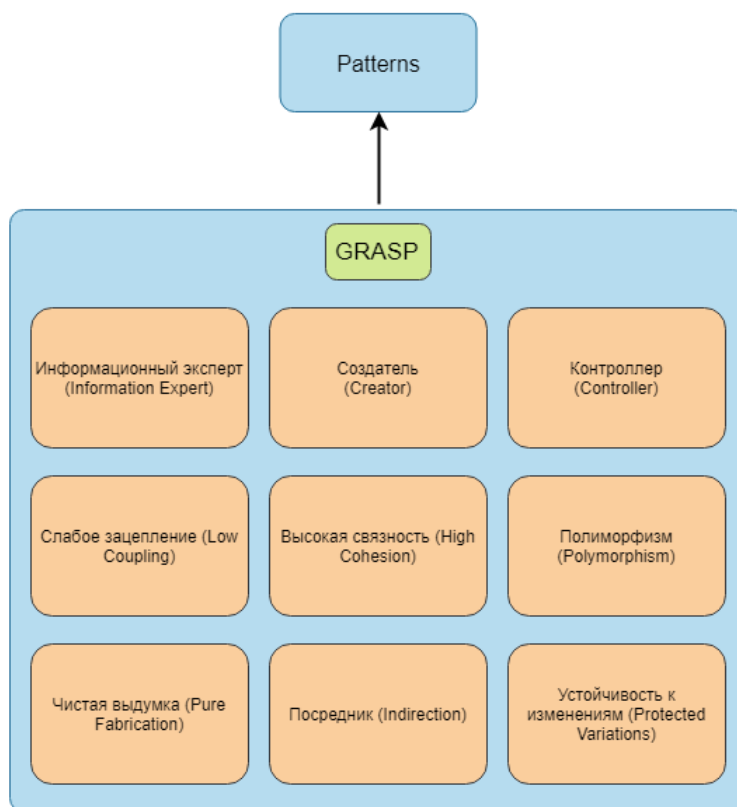


Рисунок 1 – Шаблоны GRASP

Рассмотрим каждый из основных шаблонов GRASP, их применение, а также преимущества и недостатки [1].

1. Creator (Создатель)

Что делает: шаблон Creator определяет, какой объект должен быть ответственен за создание других объектов. Он помогает управлять связями между объектами, уменьшая зависимости между ними.

Когда используется: применяется, когда есть необходимость в создании объектов, исходя из логических связей между ними.

Преимущества:

- уменьшение связанности: помогает снизить зависимости между объектами;
- повышение гибкости: упрощает добавление новых классов или изменение логики создания объектов.

Недостатки:

- сложность выбора создателя: не всегда ясно, какой объект должен создавать другие объекты.

2. Information Expert (Эксперт по информации)

Что делает: этот шаблон назначает задачу объекту, обладающему необходимой информацией для ее выполнения.

Когда используется: применяется, когда объект обладает наибольшим объемом информации, необходимой для выполнения определенной задачи.

Преимущества:

- более логичный код: задача поручается объекту с наибольшей информацией;
- уменьшение связанности: позволяет уменьшить зависимость между объектами.

Недостатки:

- сложность выбора эксперта по информации: не всегда очевидно, какой объект обладает наибольшей информацией.

3. Controller (Контроллер)

Что делает: контроллер является промежуточным объектом, координирующим действия других объектов. Он принимает запросы от пользователей и решает, какие объекты должны выполнить необходимые действия. Контроллер должен делегировать работу, которая должна быть сделана другим объектам; он координирует или контролирует деятельность [2].

Когда используется: применяется в системах, где требуется централизованное управление потоком данных и действий.

Преимущества:

- централизация управления: обеспечивает простоту координации между различными частями системы;
- уменьшение связанности: позволяет уменьшить зависимость между объектами и пользовательским интерфейсом.

Недостатки:

- возможное усложнение: в больших системах может потребоваться более сложное управление контроллером.

4. Low Coupling (Низкая связанность)

Что делает: этот шаблон направлен на уменьшение зависимостей между различными компонентами системы.

Когда используется: применяется для создания более гибкой и легко изменяемой системы.

Преимущества:

- гибкий и легко изменяемый код: уменьшение зависимостей делает систему более устойчивой к изменениям.

Недостатки:

- сложность в проектировании: требует тщательного планирования связей между объектами, что может занять больше времени.

5. High Cohesion (Высокая связанность)

Что делает: шаблон High Cohesion направлен на группировку функционально связанных элементов вместе для обеспечения их лучшей организации.

Когда используется: применяется для создания более понятного и поддерживаемого кода. Тут речь о том, что обязанности данного элемента тесно связаны и сфокусированы [3].

Преимущества:

- улучшенная читаемость и поддержка: система становится более понятной и легко поддерживаемой;
- более логичное разделение функций: объекты сфокусированы на конкретных задачах.

Недостатки:

- ограничение универсальности объектов: может потребоваться больше объектов для разделения функциональности.

6. Polymorphism (Полиморфизм)

Что делает: полиморфизм позволяет объектам с одинаковым интерфейсом иметь различное поведение.

Когда используется: применяется для создания более гибких и расширяемых систем.

Преимущества:

- гибкость и расширяемость: легко добавлять новые классы и методы без изменения существующего кода;
- упрощение общей структуры: позволяет использовать единый интерфейс для различных реализаций.

Недостатки:

- сложность в понимании: некоторые концепции полиморфизма могут быть сложными для начинающих разработчиков.

Таким образом, шаблоны GRASP представляют собой мощный инструмент для разработчиков программного обеспечения, обеспечивая четкое и логичное распределение ответственности между объектами системы. Их использование может значительно улучшить качество кода и структуру программных проектов, однако требует аккуратного подхода и понимания контекста применения для достижения максимальной пользы.

Литература

1. GRASP принципы [Электронный ресурс]. – 2019. – URL: <https://bool.dev/blog/detail/grasp-printsipy>. – Дата доступа: 15.02.2024.
2. GRASP [Электронный ресурс]. – 2019. – URL: <https://alextoolsblog.blogspot.com/2019/12/grasp.html>. – Дата доступа: 19.02.2024.
3. SOLID, GRASP и другие принципы разработки [Электронный ресурс]. – 2023. – URL: <https://sptm.dev/2023/solid-grasp-and-stuff/>. – Дата доступа: 17.02.2024.

М. М. Лакизо

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. А. Дробышевский**, ст. преподаватель

ИССЛЕДОВАНИЕ ВЛИЯНИЯ СОЦИАЛЬНЫХ МЕДИА НА ПРИВЛЕЧЕНИЕ КЛИЕНТОВ ФОТОСТУДИИ «ИП КАРАХАНОВА Ж. В.»

Исследование влияния социальных медиа на привлечение клиентов фотостудии является важным шагом в осмыслении современных маркетинговых стратегий. С учетом активного участия пользователей в социальных платформах, необходимо рассмотреть, насколько эффективно фотостудия может использовать социальные медиа для увеличения своей клиентской базы.

Анализ присутствия в социальных медиа. На первом этапе исследования были проанализированы аккаунты и электронные страницы фотостудии в социальных сетях, таких как Instagram, Facebook, ВКонтакте и Telegram, чтобы оценить, насколько широко и эффективно она представлена в онлайн-сообществах.

Взаимодействие с аудиторией. Был проведен анализ лайков, комментариев, и репостов, что помогло определить, насколько активно и эффективно фотостудия взаимодействует с потенциальными клиентами, а также понять, какие типы контента наиболее привлекательны для аудитории.

Эффективность рекламных кампаний. Исследование включает в себя анализ результатов рекламных кампаний фотостудии в социальных медиа. Была проведена оценка кликабельности рекламных объявлений, их охвата и вовлеченности, что позволило определить, насколько эффективными являются инструменты рекламы в данном контексте.

Анализ конверсий. Был проведен анализ конверсий – от перехода пользователя по рекламной ссылке до завершения сделки. Это помогло оценить степень преобразования потенциальных клиентов в реальных заказчиков через социальные медиа.

Выводы и рекомендации. На основе проведенного исследования были сформулированы выводы и рекомендации, которые включают в себя предложения по улучшению контента, оптимизацию времени публикаций, а также эффективное взаимодействие с аудиторией для создания долгосрочных отношений с клиентами.