**ГРУППА К8-224**



**СЕРВИС ПО ПОИСКУ ВЕЧЕРИНОК. МЕТОДОЛОГИЯ**

|  |
| --- |
|  |
| «PartySurfing» |

**Москва 2014 - 2015**

**Содержание**

[1. Общее описание методологии, используемой в проекте PartySurfing 3](#_Toc416724200)

[2. Основные артефакты методологии 4](#_Toc416724201)

[2.1. Список особенностей сервиса (features, далее фича) 4](#_Toc416724202)

[2.2. Бэклог спринта (Sprint backlog) 4](#_Toc416724203)

[2.3. Собрание спринта (Sprint meeting, далее спринг митинг) 4](#_Toc416724204)

[2.4. Спринт (Sprint, далее спринт) 5](#_Toc416724205)

[2.5. Жизненный цикл спринта 5](#_Toc416724206)

[2.5.1 Планирование спринта 5](#_Toc416724207)

[2.5.2 Разработка 6](#_Toc416724208)

[2.5.3 Документирование 6](#_Toc416724209)

[2.5.4 Тестирование 6](#_Toc416724210)

[2.5.5 Демо и ревью спринта 7](#_Toc416724211)

История изменений

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Версия | Изменения | Дата | Автор | Подпись |
| 1.0 | Черновик | 05.03.2015 | Бодунков Денис |  |
| 1.1 | Приведение документа к стандарту | 06.03.2015 | Македонская Евгения |  |
| 1.2 | Редактирование документа | 08.03.2015 | Македонская Евгения |  |
| 2.0 | Переработка методологии в соответствии с особенностью разработки проекта | 25.03.2015 | Македонская Евгения |  |
| 2.1 | Доработка методологии | 13.04.2015 | Македонская Евгения, Таракчян Левон |  |
|  |  |  |  |  |

# Общее описание методологии, используемой в проекте PartySurfing

Методология, используемая в проекте сервис по поиску вечеринок PartySurfing (далее методология) является гибкой методологией разработки программного обеспечения, основанной на принципах Agile. Методология включает в себя серию подходов к разработке программного обеспечения, ориентированных на использование итеративной разработки, динамическое формирование требований и обеспечение их реализации в результате постоянного взаимодействия внутри самоорганизующихся рабочих групп, состоящих из специалистов различного профиля.

В методологии, используемой для разработки сервиса сохраняются основные принципы Agile:

1. Люди и взаимодействие важнее процессов и инструментов;
2. Работающий продукт важнее исчерпывающей документации;
3. Сотрудничество с заказчиком важнее согласования условий контракта;
4. Готовность к изменениям важнее следования первоначальному плану.

# Основные артефакты методологии

## Список особенностей сервиса (features, далее фича)

Список фич - это список имеющихся бизнес-требований к системе в рассматриваемый момент времени, которые имеют различный приоритет в зависимости от важности требования.

Список фич включает необходимые к реализации возможности сервиса, то есть не включает в себя способы применения (use case), список багов сервиса (defects), технологий, используемых при разработке сервиса (technologies), историй использования (stories) и задач (issues).

По решению большинства команды разработки в список фичи могут быть добавлены новые задачи и особенности, а также удалены старые. Список фич содержит ориентировочные даты начала и конца их выполнения, которые (с учетом приоритета) служат основой для формирования и изменения проектного плана разработки сервиса. После выполнения задачи фича удаляется из списка, что обеспечивает его актуальность.

## Бэклог спринта (Sprint backlog)

Бэклог спринта содержит функциональность фичи (или фич) из списка фич, разрабатываемую в данный момент командой. Все функции разбиты по задачам, каждая из которых закреплена за ответственным. Ответственный за задачу по согласованию с менеджером проекта может взять себе помощника из числа свободных членов команды. Два раза в неделю команда собирается на спринт митинг, чтобы оценить объем работы, который нужно проделать для завершения задач.

## Собрание спринта (Sprint meeting, далее спринг митинг)

Вся команда два раза в неделю собирается на спринт митинг для обсуждения возникших проблем, задач в разработке и текущих результатов. В ходе спринт митинга обсуждаются предложения и замечания. Спринт митинг проходит по средам и субботам. При необходимости может быть назначен дополнительный спринт митинг, на который может собираться только часть команды.

## Спринт (Sprint, далее спринт)

Одна итерация называется спринт, с длительностью в одну неделю. Распределение задач по спринту происходит с учетом занятости членов команды и их специализации.

Результатом спринта является готовый продукт (build), который готов к передаче (или показу) заказчику.

В течение каждого спринта используется каскадная модель разработки, то есть производится работа по сбору требований, дизайну макета, разработке, верстке, документированию фич а также последующим их тестированием и отладке.

По результатам спринта происходит оценка работы каждого члена команды. Выполненные задачи за спринт и оценки каждого участника команды предоставляются преподавателю.

## Жизненный цикл спринта

### Планирование спринта

В начале каждого спринта проводится планирование спринта, в котором участвует вся команда.

Планирование происходит на спринт митинге в субботу. В течение митинга подробно рассматриваются задачи, а также определяются ответственные за них, помощники и определяется подробные задачи. Для каждой задачи создается подробное ишью (issue) на github, и назначается ответственный. Дальнейший жизненный цикл задачи может включать в себя (анализ, разработку, документирование и тестирование).

### Разработка

Задачи на разработку назначаются после завершения анализа или по результатам планирования спринта (при условии, что все аспекты были обговорены на спринт митинге). Задачи на этом этапе могут быть связаны с разработкой нового функционала или с изменением и/или версткой дизайна.

При разработке функционала ответственным за задачу назначается ведущий разработчик, который может перераспределять задачи в связи со специализацией членов команды и их занятостью на протяженности спринта. При изменении и/или верстке дизайна задача закрепляется за главным дизайнером, который также может их перераспределять.

### Документирование

После завершения разработки ответственным за задачу назначается технический писатель.

После завершения задач связанных с версткой и/или дизайном технический писатель ответственен за создание схем для документации.

После документирования ишью закрываются.

### Тестирование

После документирования всех задач спринта технический писатель передает техническое задание ответственному за тестированию. Ответственный за тестирование, а также его помощники проводят тестирование всех задач, выполненных на этой неделе, а так же полное тестирование сервиса. Процесс тестирования включает в себя создание тест-требований, написание тест-кейсов, ручное тестирование для новых задач и написание автоматических тестов для последующего полного тестирования на следующих спринтах.

### Демо и ревью спринта

Команда демонстрирует преподавателям инкремент продукта, созданный за последний спринт. Команда рассказывает о поставленных задачах, способах их решения, сложностях, возникших в течении спринта, принятых решениях и нерешенных задачах. На основании ревью принимающая сторона (преподаватели) может сделать выводы о том, как должна дальше развиваться система. Участники митинга делают выводы о том, как шел процесс в команде и предлагает решения по его улучшению.