

# Tools and Artifacts

**Prof. Dr. Dirk Riehle**

**Friedrich-Alexander University Erlangen-Nürnberg**

**AMOS B02**

Licensed under [CC BY 4.0 International](https://creativecommons.org/licenses/by/4.0/)

# Agenda

1. Code repository
2. Project features
3. Kanban board
4. Planning docs

# Project Management

- The AMOS Project directive is **simplify, simplify, simplify!**
  - GitHub repository, issues, and project for software development
  - Google Sheets for high-level product and process management
  - Teams may add additional tools at their discretion (and own risk)

# The Wahlzeit / Flowers Example

## Wahlzeit

[ [show](#) | [tell](#) ] — [ [signup](#) | [login](#) | [configure](#) ]

### About


This website is to show, discuss, and praise photos!

### Photo Filter

[Click to toggle filter!](#)

### Community

[It is Wahlzeit!](#)



Who/what/where is that?  
[Click to show/hide description!](#)

### Praise it!

☐ 10  
☐ 9  
☐ 8  
☐ 7  
☐ 6  
☐ 5  
☐ 4  
☐ 3  
☐ 2  
☐ 1  
Or [skip](#) it.

a friend about this photo: <http://localhost:8585/x1ac6.html> — Send  to the owner of this photo!

Please help keep this community site clean!  photo as inappropriate if necessary.

**This website is to show the best in photos!**

[ [blog](#) ] — [ [about](#) | [contact](#) | [imprint](#) | [terms](#) ] — [ language: en | [de](#) ] — [ photo size: [XS](#) | [S](#) | M | [L](#) | [XL](#) ] — [ debug: [reset](#) ]

[ processing time: 0.001 seconds ]

# 1. Code Repository

# GitHub Code Repository Set-up

- GitHub account
  - Please create a GitHub account if you don't have one yet
  - Send your account name (handle) to team coach
- GitHub project
  - The team coach will create a team for you and invite you to a code repository
  - You will all be owners of the code repository, initially; change as you see fit
- Your project name will be amos-ss20xx-fabulous-project
  - It will be public (read access to world)
  - It uses the MIT license
- You can collaborate as you see fit (but should agree first)

## 2. Project Features

# Feature and User Story

- Definition of **feature**
  - A distinguishing characteristic of a software item (for example, performance, portability, functionality) (IEEE 829)
- Definition of **user story**
  - A description mechanism for a software feature used to create a shared understanding of the feature; follows an established pattern
  - The description pattern is: As a “**user role**”, I need “**a function**”, so that “**I get business value**” where the parts in quotation marks are substituted
- Example user stories
  - Rate-a-photo: As a **visitor**, I can **rate a photo with a numerical value 1-10**, so that **I can share my opinion about the photo**.
  - Tell-a-friend: As a **user**, I need **a function to tell a friend about a flower photo**, so that **I can share my passion for flowers**.



# Types of Issues

- Features
  - New functional improvements
- Refactorings
  - Behavior-preserving improvements
- Bugs
  - Fixes to malfunctioning

# Quality Criteria for Features

**I** ndependent: Features should be independent of each other.

**N**egotiable: A feature is man-made, not cast in stone.

**V** aluable: Every feature should have business value.

**E** stimatable: A feature should be precise so that size can be estimated.

**S** mall: A feature should be small enough for one iteration.

**T** estable: A feature should have testable success criteria.

# GitHub Issues Set-up

- GitHub issue template
  - Please create a template for feature requests
  - The template should contain fields for
    - Short name (already preset as title field)
    - Short description (using user story)
    - Acceptance criteria (to test for fulfillment)
    - Definition of done (from week 5 on)
- Create labels for features sizes
  - Est. size = X for estimated sizes
    - Set during sprint planning
  - Real size = Y for actual size
    - Set during sprint review
  - Use escalating colors
- Feel free to add more templates

## Issue: Feature Request

Suggest new feature (using user story)

### User story

1. As a {},
2. I want / need {}
3. So that {}

### Acceptance criteria

- Criterion 1
- Criterion 2
- ...

### Definition of done

- Added after week 5
- Always the same

### Optional additional items

Issue default title:

Assignees:

Labels:

# The Tell a Friend Feature 1 / 2



**Tell a friend #129**

dirkriehle opened this issue 2 days ago · 0 comments



**dirkriehle** commented 2 days ago • edited ▾

Owner



## User story

1. As a guest
2. I can email a friend about a photo
3. So that I can share my passion

## Acceptance criteria

- After hitting OK, an email is sent
- After hitting Cancel, no email is sent
- The next page is the main page

## Definition of done

To be defined week 5 onwards.




# The Tell a Friend Feature 2 / 2



[ [show](#) | [tell](#) ] — [ [signup](#) | [login](#) | [configure](#) ]

## Tell a friend!

	From Email Address:	<input type="text" value="max@muster.com"/>
	To Email Address:	<input type="text" value="mara@other.org"/>
	Email Subject:	<input type="text" value="Photo sharing website!"/>
	Message:	<div><div>Checkout this site!</div><div><a href="http://localhost:8585/">http://localhost:8585/</a></div><div>In particular, this photo!</div><div><a href="http://localhost:8585/x1ac6.html">http://localhost:8585/x1ac6.html</a></div></div>
		<input type="button" value="Tell!"/>

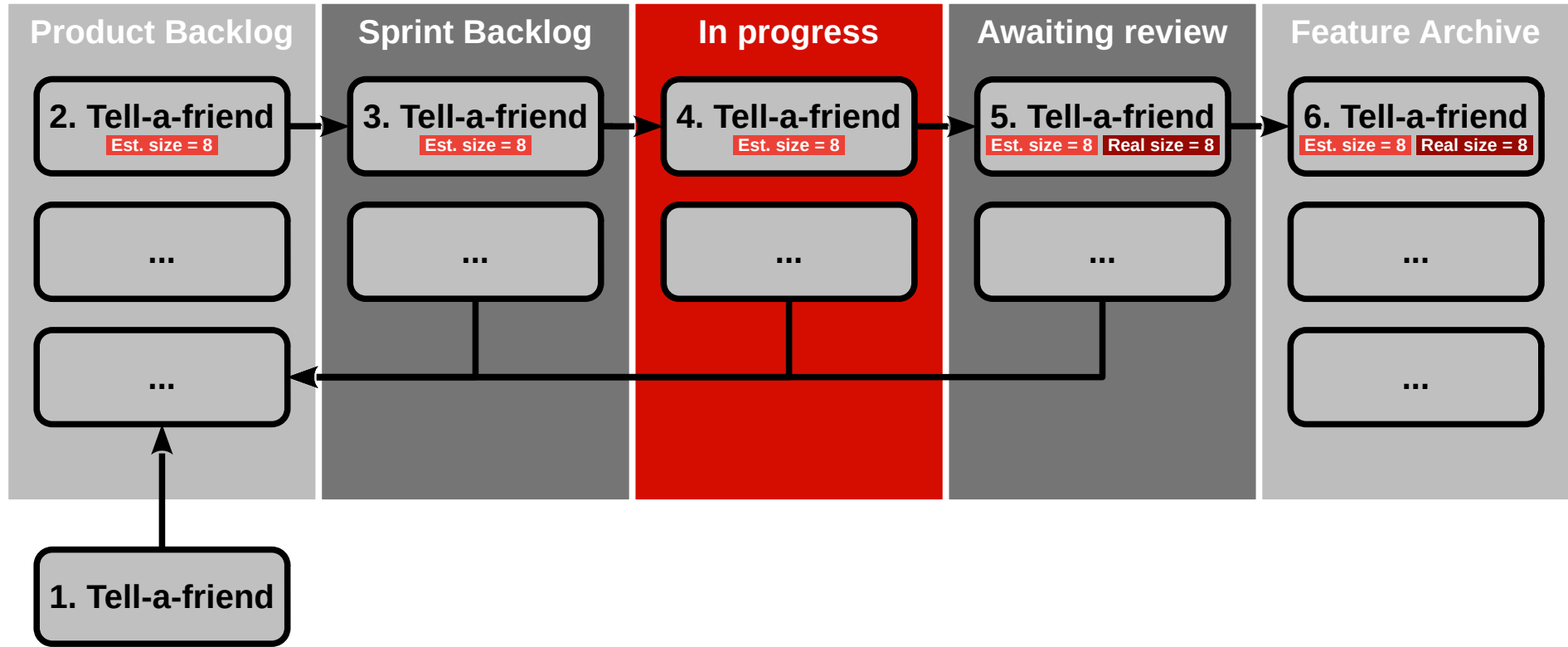
**This website is to show the best in photos!**

[ [blog](#) ] — [ [about](#) | [contact](#) / [imprint](#) | [terms](#) ] — [ language: en | [de](#) ] — [ photo size: [XS](#) | [S](#) | M | [L](#) | [XL](#) ] — [ debug: [reset](#) ]

[ processing time: 0.005 seconds ]

## 3. Kanban Board

# The Feature Life-Cycle



# Product Backlog and Product Backlog Entry

- **Product backlog entry**
  - A named, identifiable, and prioritized feature, bug, or refactoring
- **Product backlog**
  - An incomplete, evolving list of features prioritized by business value
- Purpose of product backlog
  - To know what's coming up next



# Sprint Backlog and Sprint Backlog Entry

- **Sprint backlog entry**
  - A high-quality product backlog entry that
    - has been estimated for size and committed-to to being implemented by the software developers
- **Sprint backlog**
  - A list of sprint backlog entries that
    - the software developers have committed to implementing in the current sprint
- Purpose of sprint backlog
  - To know at any time what to do in the current sprint
  - To serve for task allocation between developers

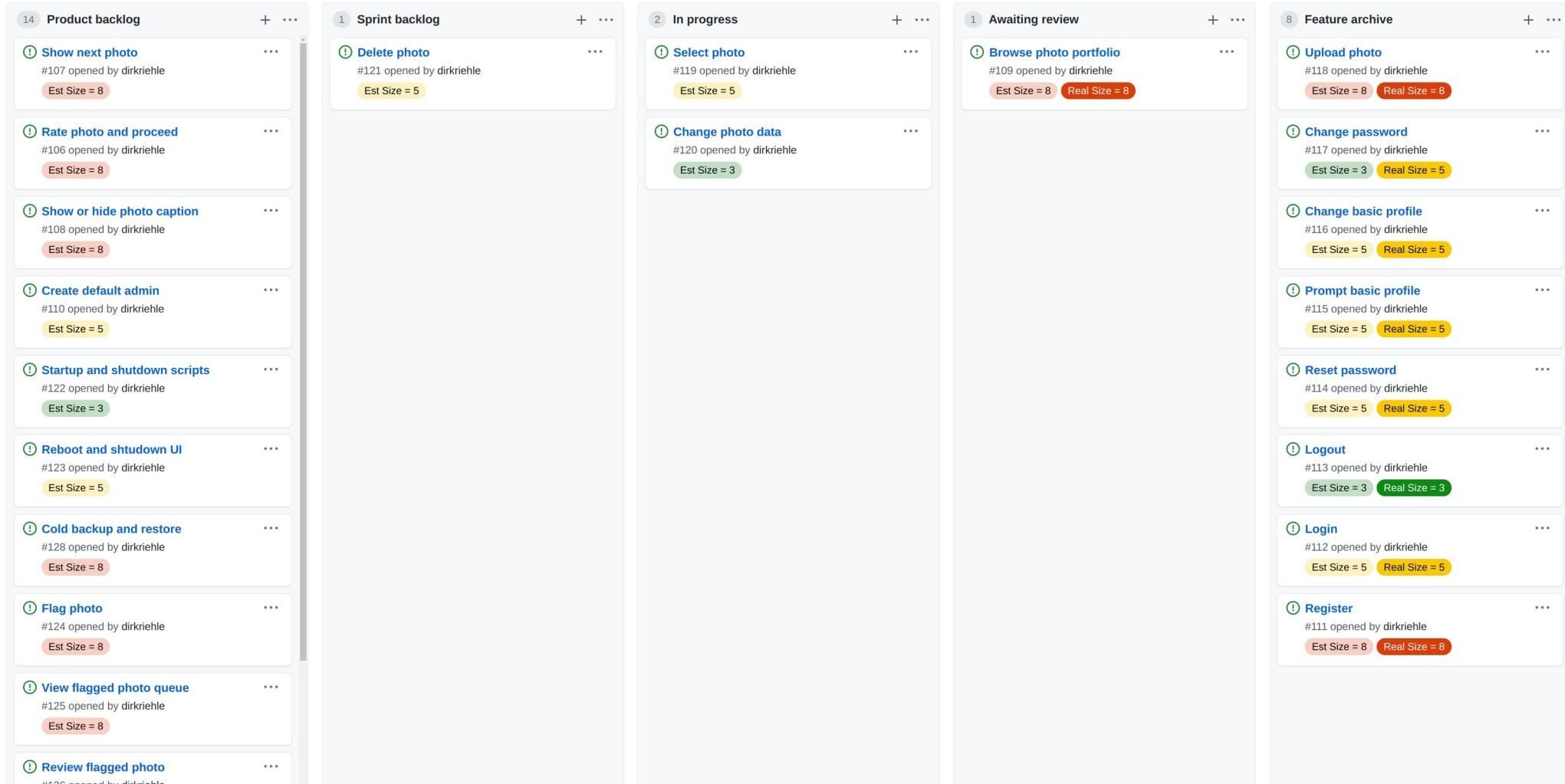
# Feature Archive

- **Feature archive**
  - All issues that have been implemented
  - All issues contain the original size estimation
  - All issues contain the real (after-implementation) size
- Purpose of feature archive
  - To serve as a summary of product features
  - To allow lookup of history of development


# GitHub Project Set-up

- GitHub project
  - Please create a project (board) for backlogs
  - The board should have five columns
    1. Product backlog (for new unscheduled features)
    2. Sprint backlog (for features to-be-done)
    3. In progress (for features currently in work)
    4. Awaiting review (for features to be reviewed)
    5. Feature archive (for finished features)
- Connect issues with project




# The Wahlzeit / Flowers Kanban Board







# Task Planning / Assigning Backlog Entries



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

 [dirkriehle](#) / [wahlzeit](#)

 Unwatch 4  Star 5  Fork 289

[Code](#) [Issues 34](#) [Pull requests 1](#) [Actions](#) [Projects 1](#) [Security](#) [Insights](#) [Settings](#)

### Wahlzeit Backlogs

Updated on Mar 6

14 **Product backlog** + ...

Show next photo

#107 opened by dirkriehle

Est Size = 8

Rate photo and proceed

#106 opened by dirkriehle

Est Size = 8

Show or hide photo caption

#108 opened by dirkriehle

Est Size = 8

Create default admin

#110 opened by dirkriehle

Est Size = 5

1 **Sprint backlog** + ...

Delete photo

#121 opened by dirkriehle

Est Size = 5

2 **In progress**

Select photo

#119 opened by dirkriehle

Est Size = 5


Change photo data

#120 opened by dirkriehle

Est Size = 3

Delete photo #121

Opened in dirkriehle/wahlzeit


 **dirkriehle**  
commented on Mar 4


As a user, I can select any photo from my portfolio and have it deleted

After deleting the photo, it will not be shown any longer to any user

0

Assignees

 **dirkriehle**

 **georg-schwarz**

Labels

## 4. Planning Documents

# Strategic Planning Set-up

- Strategic planning uses Google Sheets
  - Please copy the planning documents template: <http://goo.gl/Brc0Q>
  - Share your copy with the team and the coach
- Wahlzeit / Flowers example
  - Planning artifacts: <http://bit.ly/3qitBF1>

# Strategic Planning Document

#	Artifact Name	Artifact Purpose
1	Project Data	Provides basic project data
2	Project Team	Shows project team and static roles
3	Team contract	Shows team contract
4	Role Assignments	Tracks role assignment
5	Product Goal	Provides product vision and project mission
6	Product Glossary	Provides domain terminology of project
7	Mid-project Release Tracking	Tracks mid-term release sprints
8	Final Project Release Planning	Plans and tracks final project release sprints
9	Impediments Backlog	Tracks process problems and solutions
10	Definition of Done	Provides decision criteria for “done”
11	Documentation	Provides additional documentation of product
12	Bill of Materials	Lists all third-party components
13	Planning Poker	Provides simple tool for planning poker



# 1. Project Data

Project Name	Flowers (Wahlzeit)
Production system (if any)	<a href="https://wahlzeit.com/flowers">https://wahlzeit.com/flowers</a>
Test system (if any)	...
Github repository	<a href="https://github.com/dirkriehle/wahlzeit/">https://github.com/dirkriehle/wahlzeit/</a>
GitHub kanban board (project)	<a href="https://github.com/dirkriehle/wahlzeit/projects/2">https://github.com/dirkriehle/wahlzeit/projects/2</a>
Team T-shirt (white)	...
Team T-shirt (black)	...
Additional materials	...

# 1. Project Data (Explanation)

- Definition
  - A collection of relevant (starting point) links
- Purpose
  - To simplify navigation
- Production and system
  - Deployment scenarios depend on project
- GitHub repository
  - Required
- CI tooling
  - Optional

## 2. Project Team

Last Name	First Name	GitHub User Name	Email Address
Riehle	Dirk	dirkriehle	dirk.riehle@fau.de
Schwarz	Georg	georg-schwarz	georg.schwarz@fau.de
Bauer	Andi	andreas-bauer	andi.bauer@fau.de
Hahmann	Lukas	tfrdidi	lukas.hahmann@fau.de
Wagemann	Hendrik	Henny022	hendrik.wagemann@fau.de
Metzig	Aron	knusperkrone	aron.metzig@fau.de
Dohrn	Hannes	hannesd	hannes.dohrn@fau.de

## 2. Project Team (Explanation)

- Definition
  - The list of team members and their contact information
- Purpose
  - To make it easy to contact individuals or the whole team
- Pragmatics
  - Feel free to add other information, e.g. Skype ID

# 3. Team Contract

<b>Goals</b>	Achieve our defined project mission
	Foster and atmosphere of learning
<b>Meeting norms</b>	Everyone shows up on-time
	We do not interrupt each other
<b>Working norms</b>	Everyone contributes regularly
	We value quality over quantity
<b>Coordination norms</b>	Every job has a responsible person
	We volunteer for jobs (no assignment)
<b>Communication norms</b>	We check Slack at least once a day
	We follow the Chatham house rules
<b>Consideration norms</b>	We discuss disagreement openly
	We vote for a final resolution
<b>Cont. improvement norms</b>	We jointly review the happiness index
	You must raise insufficient quality issues
<b>Rewards</b>	We high-five a successful sprint release
	We celebrate a successful release with cookies
<b>Sanctions</b>	You must raise clear violations of the team contract
	We apologize by bringing cookies

### 3. Team Contract (Explanation) [1]

- Definition
  - A uniform agreement of the team as to which goals, norms, rewards, and sanctions they play by
- Purpose
  - To have a defined and agreed-upon way of operating as a team

[1] <https://oss.cs.fau.de/teaching/course-resources/student-team-contract/>

## 4. Role Assignments

#	Meeting Day	Comment	Coach	Product Owner	Software Developer	Scrum Master	Release Manager
1	2021-04-14			Dirk Riehle	Everyone else	N/A	N/A
2	2021-04-21			Dirk Riehle	Everyone else	Dirk Riehle	Dirk Riehle
3	2021-04-28			Dirk Riehle	Everyone else	Dirk Riehle	Dirk Riehle
4	2021-05-05			Dirk Riehle	Everyone else	Dirk Riehle	Dirk Riehle
5	2021-05-12			Dirk Riehle	Everyone else	Dirk Riehle	Dirk Riehle
6	2021-05-19			...	...	...	...
7	2021-05-26	Mid-term due					
8	2021-06-02						
9	2021-06-09						
10	2021-06-16						
11	2021-06-23						
12	2021-06-30						
13	2021-07-07						
14	2021-07-14	Demo day!					
15	2021-07-21	Retrospective					

## 4. Role Assignments (Explanation)

- Definition
  - A week-by-week assignment of roles to team members
- Purpose
  - To have a plan and history of role assignments so everyone knows their job
- Pragmatics
  - Please plan out the project at start of course
  - Feel free to add roles beyond the four required ones
- **Use the same week numbers as in the Course schedule**
  - **Week 1 starts with the first lecture**
  - **Week 1 also starts the first sprint**



# 5. Product Goal

Product Vision	Project Mission
<p>The Flowers social network helps flower enthusiasts world-wide to connect with each other and enjoy following their favorite hobby online. Centered on showing and rating favorite flower photos, it inspires growing and presenting ever more beautiful flowers. With a highly engaged user community, Flowers is the best place for producers and sellers of gardening supply to reach out to customers and engage with them. Such engagement involves understanding flower enthusiasts' needs around gardening supplies and selling to them.</p>	<p>The mission of this project is to create an MVP for Wahlzeit with the Flowers extension. Core functionality will be showing and rating photos, basic user management, case management, and minimal system administration.</p>

## 5. Product Goal (Vision and Mission)

- Definition of product vision
  - A product vision communicates
    - The main purpose and core functionality of the product
    - The value the main stakeholders receive
  - It is inspiring, timeless, focused and the final arbiter
- Definition of project mission
  - A project mission summarizes
    - What the project wants to achieve
    - Within the project time-frame
- Purpose
  - To succinctly explain product and project
  - To guide when prioritizing features

## 6. Product Glossary

Term	Definition
Average Photo Rating	Synonym for overall photo rating.
Individual Photo Rating	A numerical value 1..10 that a visitor applies to a photo.
Overall Photo Rating	A photo rating is a numerical value that represents the community's ranking of the photo in question. Over time, a photo accumulates a large number of individual photo ratings.
Photo	A photo is a flower photo that has been uploaded (and is owned) by a system user. A photo has a status that determines its accessibility.
Photo Sharing Policy	Flowers has a photo sharing policy. The policy determines what an acceptable photo is. Photos are unacceptable if they are unrelated to the current flower.
Photo status	A photo may be visible to the world or invisible (by choice of owner). Photos may be flagged for violating Flowers' photo sharing policies. Flowers moderators can change the status of a photo.
Photo Tag	A photo tag is a label that has been applied to a photo by its owner. They have no further semantics beyond the actual text.
Single Photo Rating	Synonym for individual photo rating.
System Administrator	A system administrator is a system moderator with additional administration functionality. He or she can reboot the system.
System Moderator	A system moderator is a system user with access to additional moderation functionality. He or she can moderate photos.
System User	A system user is a (human) user of the system who has a user account and is logged into that account. A user can upload and maintain photos.
System Visitor	A system visitor is (human) user of the system. A visitor has access to all public functionality that does not require a login.
Tag	Shorthand for photo tag.
User	Shorthand for system user.

## 6. Product Glossary (Explanation)

- Definition
  - Is an alphabetically ordered list of terms and their definition
- Purpose
  - To provide key domain terminology (shared vocabulary) of project
  - Is a simplified form of a domain model accessible to everyone
- Term
  - A (possibly qualified) noun determining the concept to be defined
- Term definition
  - The definition of the referenced term
  - A base definition is a declarative statement of the form X is a X\*
  - Other forms of definitions are synonyms, short-hands, etc.

# 7. Mid-Project Release Tracking

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
1	Visitor	Self-Admin			21		21	
			<b>Deliver first increment of running software</b>					
			Register	8		8		
			Login	5		5		
			Logout	3		3		
			Reset Password	5		5		
2	User	Self-Admin			21		23	
			<b>Deliver increment with basic user handling</b>					
			Prompt Basic Profile	5		5		
			Change Basic Profile	5		5		
			Change Password	3		5		
			Upload Photo	8		8		
3	Photo	Management			21			
			<b>Deliver increment with basic photo handling</b>					
			Browse Photo Portfolio	8				
			Select Photo	5				
			Change Photo Data	3				
			Delete Photo	5				
4	...							

## 7. Mid-Project Release Tracking (Explanation)

- Definition
  - A prioritized list of features for a product release
- Purpose
  - To predict the content of the upcoming product release
- Pragmatics
  - Has been estimated by developers, validated by historic data
  - Is a living document, that is, gets adjusted after each sprint

## 8. Final Project Release Planning and Tracking

[illegible]

## 8. Final Project Release Planning and Tracking (Explanation)

- Like mid-project release tracking, but with added planning



## 9. Impediments Backlog

[illegible]

## 9. Impediments Backlog (Explanation)

- Definition
  - A list of current and historic impediments to process
- Purpose
  - To know at any time the current impediments
  - To see whether any impediments keep recurring
- Pragmatics
  - It names the Scrum master responsible for resolution
  - It provides the solution found, if any

## 10. Definition of Done

[illegible]

# 10. Definition of Done (Explanation)

- Definition
  - Is a check list of decision criteria to determine status of “done”
- Purpose
  - To be consistent about what “done” means
- Pragmatics
  - Definitions of “done” (DoD) require team consensus
  - There are many such decisions, hence many DoDs needed
  - At a minimum, at the proper time, the team should have a
    - DoD for features
    - DoD for sprint release
    - DoD for product release

# 11. Documentation

Type	Link / reference
User manual	<a href="https://github.com/dirkriehle/wahlzeit/docs">https://github.com/dirkriehle/wahlzeit/docs</a>
Developer documentation	<a href="https://github.com/dirkriehle/wahlzeit/docs">https://github.com/dirkriehle/wahlzeit/docs</a>

# 11. Documentation (Explanation)

- Definition
  - The product documentation (where to find what)

## 12. Bill of Materials

#	Context	Name	Version	License	Comment
1	com.fasterxml.jackson.core	jackson-annotations	2.9.5	Apache-2.0	
2	com.fasterxml.jackson.core	jackson-core	2.9.5	Apache-2.0	
3	com.fasterxml.jackson.core	jackson-databind	2.9.5	Apache-2.0	
4	com.fasterxml.jackson.dataformat	jackson-dataformat-xml	2.8.10	Apache-2.0	
5	com.fasterxml.jackson.dataformat	jackson-dataformat-yaml	2.9.5	Apache-2.0	
6	com.fasterxml.jackson.datatype	jackson-datatype-guava	2.9.5	Apache-2.0	
7	com.fasterxml.jackson.datatype	jackson-datatype-jdk8	2.9.5	Apache-2.0	
8	com.fasterxml.jackson.datatype	jackson-datatype-joda	2.9.5	Apache-2.0	
9	com.fasterxml.jackson.datatype	jackson-datatype-jsr310	2.9.5	Apache-2.0	
10	com.fasterxml.jackson.jaxrs	jackson-jaxrs-base	2.9.5	Apache-2.0	
11	com.fasterxml.jackson.jaxrs	jackson-jaxrs-json-provider	2.9.5	Apache-2.0	
12	com.fasterxml.jackson.module	jackson-module-afterburner	2.9.5	Apache-2.0	
13	com.fasterxml.jackson.module	jackson-module-jaxb-annotations	2.9.5	Apache-2.0	
14	com.fasterxml.jackson.module	jackson-module-parameter-names	2.9.5	Apache-2.0	
15	com.fasterxml.woodstox	woodstox-core	5.0.3	Apache-2.0	
16	...	...	...	...	
17					
18					
19					

## 12. Bill of Materials (Explanation)

- Definition
  - A list of items (“materials”) included in the project
  - Items typically mean open source libraries
- Purpose
  - To maintain an overview of “what’s the box”
- Pragmatics
  - Please lists all third-party components used by the product
  - List only those used directly; no need to recursively unpack
  - In case of open source components, also list their license



# 13. Planning Poker

Last Name	First Name	Value					
Riehle	Dirk	5		4.29	NOK		
Schwarz	Georg	2					
Bauer	Andi	5					
Hahmann	Lukas	5					
Wagemann	Hendrik	5		0	No effort		
Metzig	Aron	5		1	Minimal effort		
Dohrn	Hannes	3		2	Small effort		
				3	Medium effort		
				5	Large effort		
				8	Very large effort		
				13	Too large effort		

# Weekly Checklist for Artifacts

- Software product
  - Tagged, testable, deployable software code base and product
- Planning documents
  - Clean kanban board, release planning, additional information
- Process improvement
  - Impediments documented; happiness index survey taken

# Thank you! Questions?

[dirk.riehle@fau.de](mailto:dirk.riehle@fau.de) – <http://osr.cs.fau.de>

[dirk@riehle.org](mailto:dirk@riehle.org) – <http://dirkriehle.com> – [@dirkriehle](#)

# Credits and License

- Original version
  - © 2021 Dirk Riehle, some rights reserved
  - Licensed under [Creative Commons Attribution 4.0 International License](#)
- Contributions
  - None yet