

**Word template
Project report**

Study program

Software Design

TH Aschaffenburg

21.06.2025

Authors:

First name: Aadit

Matriculation number: 22 76 878

First name: Antonio

Matriculation number: 22 92 339

First name: Beshoy

Matriculation number: 22 83 841

Team name: Team 14

ASCHAFFENBURG UNIVERSITY OF TECHNOLOGY

FACULTY OF ENGINEERING AND COMPUTER SCIENCE

WÜRZBURGER STRASSE 45

D-63743 ASCHAFFENBURG

Table of contents

1 PROJECT APPROACH	5
1.1 SCRUM flow	5
1.2 Procedure for SCRUM events	5
1.2.1 Sprint	5
1.2.2 Daily Scrum	6
1.2.3 Sprint Planning 1 and 2	6
1.2.4 Review	7
1.2.5 Retrospective	7
1.3 SCRUM roles	8
1.4 DoD	8
1.5 Mapping for individual performance	8
2 TEAM	9
2.1 Team name and members	10
2.2 Team Commitment	10
2.3 Team values	10
2.4 Retrospectives	11
3 SUMMARY	12

1 Project approach (Written by Beshoy)

1.1 SCRUM flow

As a starting point: all the events related to agile (not the development) took place at the lecture room C3 - 103-102

We start with the planning; where we set the sprint goal in an understandable for every team member way.

Then we decide on what can be done to achieve it and format it in a user story way lead by the project owner and reviewed then agreed upon by the development team.

We then do a magic estimation for them, prioritize them, and the scrum master asks the team memeber to pull the user stories, some of these user stories then should have a strict deadline for how important they are.

After the first week of the sprint, the weekly scrum takes place where the team members say what they have done, what they will do and what obstacles they faced, depending on what problem they have faced and what was not finished as a result, a refinement can be done in this case.

At the end of the sprint a review and retrospective is done.
Communication each 1-2 days was done in any possible way.

1.2 Procedure for SCRUM events

Describe the SCRUM events that you have carried out in the team. In doing so, go into the practiced procedure and the most important goals and timeboxing:

1.2.1 **Sprint**

Description:

In my opinion working in sprints help break down the project into small pieces (sprints), then we break these sprints (user stories), and iterate. This helps us work organized and make it easy to achieve our goals.

The further we went into sprints the better and faster our team has progressed, it was exponential.

On average a user story would be broken down into 2-3 tasks

1.2.2 Daily Scrum

Description:

Since the 2nd sprint, our entire team has consistently participated in a weekly Scrum meeting held every Wednesday. These meetings typically lasted about 15 minutes and followed the three key questions:

- What did you accomplish this week?
- What will you work on next week?
- What obstacles are you facing?

Full team participation was ensured during these sessions. In addition to the weekly meetings, daily online communication was encouraged and soon became a habit, with team members automatically updating each other every 1-2 days.

By the 3rd sprint, the team had developed a strong culture of openness and collaboration.

One of the greatest benefits of this regular exchange was the valuable honest but respectful technical feedback shared and the team members reaching out for help.

1.2.3 Sprint Planning 1 and 2

Description:

In our experience, the greatest benefit of agile release planning was crafting the user stories, and pulling the work so we are able to start once we release the sprint plan document.

- **Sprint 1:** Setting up the virtual machine, getting the pipeline ready to accept jobs, figuring out the goals of the project.
- **Sprint 2:** Running the same security checks made for the pipeline locally and achieve the results.
- **Sprint 3:** CI/CD pipelines implementation of security tools for both of the apps in gitlab server.
- **Sprint 4:** Finish the pipeline development since some security checks were not complete and faced with issues, and start developing the dashboard (setu up the project for pulling the reports from the gitlab artifacts API)
- **Sprint 5:** the documentation is done for the project, the dashboard shows the reports and meets the deliverables.

Cutting user stories: first the user story where searched on a technical level then split into achievable tasks.

Acceptance Criteria: Acceptance criteria were decided on before the start of the user story, by the project owner with the developer or the executor of the corresponding user story.

Changes made in the planning: After the first 2 sprints, we added the set priority for each user story, and to try to set deadlines for certain tasks.

1.2.4 Review

In my opinion the greatest benefit: The team communicates what type of issues they are facing; this provides space for technical knowledge exchange, we then can solve a problem that a certain member faced for few hours in few minutes.

As the sprints go, the user stories get bigger and harder, the acceptance rate was not going up but slightly down (more work was done nevertheless), and after each weekly scrum we would refine our backlog.

By the end of the 3rd sprint the pipeline was running the checks.

1.2.5 Retrospective

Description:

In my opinion addressing what went not so well is the most important point, it helps us each address what problems we faced (agile wise) and we solve them, that helps us improve as a team by eliminating our mistakes.

The deadlines and the documentation was something that was mentioned during our retrospective, by time more deadlines were met, and documentation was done.

The documentation and the backlog are kept more clean and up to date.

It does not change; however, by time it became more easy for the team to communicate what did not go well and what we need to improve.

1.3 SCRUM roles

Scrum master: ensures the team participates and each of them give feedback and contribute to any on going process. Protect the team from the product owner. Moderates meetings, motivates the team.

Product owner: sets the goal and the vision for the product sets the acceptance criteria. Craft the user stories (then reviewed by the team).

Security Champion : member who acts as the bridge between developers and security. Their responsibilities include Advocating for security best practices in sprint planning and retrospectives as well as reviewing and maintaining security scanning tools

1.4 DoD

Insert your DoD here: <https://ahuesag.atlassian.net/wiki/x/I4AB>

Tasks are reviewed by peers, product owner accepts according to their acceptance criteria.

1.5 Mapping for individual performance (By all team members)

Antonio Guardiola:

Generic:

- Checked docker was installed in VM
- Configure gitlab-runner in VM to run pipeline jobs from gitlab projects

ToDoList's pipeline:

- Custom Image via Dockerfile
- Build stage (build image and build project)
- Publish stage (publish package and container)
- Container scanning stage
- Testing stage (unit testing and integration testing)
- Packaging stage
- Fuzzing stage

JuiceShop's pipeline:

- Custom Image via Dockerfile
- Build stage (build image and build project)
- Publish stage (publish container)

- Container scanning stage
 - Fuzzing stage
- Vulnerability dashboard:
- Chose project's stack
 - basic blank application set up
 - Settings section
 - Fuzzing section
 - Container scanning section
 - User interface

Aadit Karnavat:

Generic:

- Created folders like reports/ manually when needed to ensure artifacts existed

Both pipelines:

- Implemented the Dependency scanning
- Manually verified pipeline success

Vulnerability Dashboard:

- Developed the Dependency scanning part

Beshoy Farag:

Generic:

- Freed some space on the VM
- Connected gitlab to the virtual machine

ToDoList's pipeline:

- Implemented the SAST SonarQube
- Implemented the GitLeaks
- Did the deploy stage
- Implemented the DAST Stage
- Added the dockerhub login data for authentication for pulling docker Images

JuiceShop's pipeline:

- Implemented the SAST semgrep
- Implemented the GitLeaks
- Did the deploy stage
- Implemented the DAST Stage
- Added the dockerhub login data for authentication for pulling docker Images

Vulnerability dashboard:

- Implemented the SAST stage
- Implemented the DAST stage

2 Team

2.1 Team name and members (Written by Beshoy)

Team 14, GitGuradians

All team members participated in the backlog refinement the magic estimation and the documentation of single tasks.

Aadit Karnavat (security champion): Ensures security practices are considered, Helps identify security risks during sprint planning.

Antonio Guardiola (product owner): Product owner: sets the goal and the vision for the product sets the acceptance criteria. Craft the user stories (then reviewed by the team).

Beshoy Farag. (scrum master): ensures the team participates and each of them give feedback and contribute to any on going process. Protect the team from the Product owner.

2.2 Team Commitment (Written by Aadit)

Transparency:

Reach out for help when you need it, let the team know if there will be any delay.

Respect & communication:

Communication every 1-2 days online, and attending each lecture.

When a team member does not agree on something they should address Respectfully.

Continuous improvement:

After each sprint during the retrospective each team member should say What we can improve, the team members then follow these suggestions The next sprints.

Shared goal:

Each team member knows what our clear final goal is, each sprint Planning addresses the goal of the sprint in an understandable way

2.3 Team Values (Written by Antonio Huesa Guardiola)

For our team the most important SCRUM values have been focus, courage and commitment.

- Focus, because we must not lose our concentration on our projects' development if we want to get them into a finished state.

- Courage, because due to the little size of our team, we need the courage and effort to do the work of multiple roles and learn new things for the completion of the development.
- Commitment, because due to the complexity and challenge of the tasks, we have to ensure that the team is committed to the final goal of the projects.

2.4 Retrospectives (Edited by Aadit)

Sprint 4 Retrospective

Date: [18-06-2025]

What went well:

1. The team successfully completed the sprint work.
2. Team members supported one another by asking for help and clarifications.
3. All security tools were completed ahead of time.

What didn't go so well:

1. Time was spent on tasks that should have been completed in the previous sprint.

What needs to improve:

1. Optimize pipeline efficiency.
2. Use timeboxing and estimation for better task management.

Sprint 3 Retrospective

Date:[04-06-2025]

What went well:

1. Communication within the team improved.
2. Higher volume of completed work.
3. Greater technical understanding was achieved.

What didn't go so well:

1. Dependency management was lacking.
2. Deadlines were not clearly defined.
3. Insufficient documentation.

What needs to improve:

1. Better and more consistent documentation.
2. Start work earlier in the sprint.
3. Evaluate and plan tools before implementing.
4. Set and adhere to task timelines.
5. Actively seek feedback during development.

3 Summary (Edited by Antonio Huesa Guardiola)

All agile events were held in room C3 - 103-102. Each sprint began with planning, where goals were set for the corresponding sprint, user stories defined, estimated, and prioritized. Team members then pulled tasks (one of the scrum values not to push tasks), with some given strict deadlines for crucial user stories. Mid-sprint, a weekly scrum addressed progress and blockers, leading to refinements if needed. Each sprint ended with a review and retrospective. The team communicated regularly every 1–2 days mostly online.

The team used structured work into sprints, gradually improving in speed and quality. Sprint planning focused on defining user stories, setting goals, and assigning priorities. Weekly Scrums and regular communication supported progress tracking and collaboration. Communication helped solve technical issues quickly, while retrospectives focused on identifying problems and improving team practices. Over time, the team became more organized, met more deadlines, and maintained better documentation.

The scrum master ensures each member participates and moderates the meeting, protect the team from the product owner. The product owner defines the vision and sets the user stories and its acceptance criteria.

The team values transparency by openly asking for help and informing each other when there are delays. Communication happens on each lecture as well as each 1-2 days online. Any disagreement is handled with respect. Continuous improvement is made from retrospectives. Everyone works towards a shared goal with focus, commitment and courage, being these the three most important scrum values for our team.

4 Appendix

Unfortunately, our team was formed later in the project phase, and as a result, we were not present for the lecture covering the Lego model

formulation. We sincerely apologize for not being able to provide a Lego model for this assignment.

High Priority:

Sprint 3:

Documentation, do it and be consistent

Start earlier with the pulled tasks

Sprint 4:

Refine and check the pipeline

Mid Priority:

Sprint 3:

put time line for the task

Sprint 4:

Do timeboxing

Low Priority: