

Software Technology 2024

Γαβριήλ Κοσκινάς¹, Μιχαήλ Αβαγιανός², Θανάσης Ράτσικας³

¹Π2019211, ²inf2021009, ³inf2021193

Abstract

Abstracts must be able to stand alone and so cannot contain citations to the paper's references, equations, etc. An abstract must consist of a single paragraph and be concise. Because of online formatting, abstracts must appear as plain as possible.

General Information

Introduction

The introduction introduces the context and summarizes the manuscript. It is importantly to clearly state the contributions of this piece of work.

Software Lifecycle

Software Lifecycle or Software Development Lifecycle (SDLC) is a process used by the software industry to design, develop and test high-quality software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

Agile Methodology

Agile methodology is a type of project management process, mainly used for software development, where demands and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customers. It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change and includes six phases: Planning, Design, Development, Testing, Deployment, and Review.

Releasing to the public

Using the Agile methodology, it requires adjustment to some of the phases mentioned above. The 6-phase cycle will be repeated for each new version.

Planning

During this phase, the team will research the market and the competition to understand the needs of their customers, alongside planning the features and the timeline of the project.

Design

During the Design phase, they will create the app's design, which includes the user interface and user experience, accounting for the information they gathered through research during planning.

Development

In this phase, the team will focus on implementing the features that were planned and designed, while following certain software design principles that allow for easy implementation of new features, patching and bug fixing.

Testing

In the Testing phase, the team tests the features they implemented, and make sure they work as intended, using various testing principles such as unit testing and end-to-end testing. Unit and E2E (end-to-end) testing are automated through use of testing frameworks. Alongside them, they can incrementally test the software with real users, using A/B testing (non opt-in) and beta application clients (opt-in), to gather feedback and improve the software.

Deployment

In the Deployment phase, the team should release the software to the public, allowing for use of testing principles mentioned above, to ensure the software is working as intended.

Review

Finally, during Review, the team will gather feedback from the users, and use it to improve the software, by adding new features, patching bugs, and improving the user experience.

Data

Analysis

Results

Conclusion

Contributions

- Γαβριήλ Κοσκινάς implemented data fetching, machine learning processing and analysis.
- Μιχαήλ Αβαγιανός created the user interface, the data visualization, the UML and modularized the code.
- Θανάσης Πάτσικας containerized the application, and wrote this document.