

Engagis App Engine



1 Engagis App Engine & Agile approach

1.1 Overview

Engagis uses the AGILE development methodology SCRUM as the preferred approach for delivering all software solutions.

In a world of constantly changing requirements SCRUM delivers the lowest risk, highest value and the fastest way to respond when developing software.

Our staff are trained in SCRUM and are passionate about delivering software the “Engagis way” through our App Engine. We use cloud based software, called JIRA, to manage the requirements for software projects. JIRA has been customised to support our AGILE development approach.

AGILE development grew in response to THE challenges and limitations of traditional waterfall development practices. Traditional software development approaches have high failure rates but SCRUM counteracts this trend. It provides a framework for delivering requirements incrementally, whilst maintaining a focus on the rapid delivery of business value in a collaborative and customer focused approach.

AGILE development accelerates the delivery of initial MVP (Minimum Viable Product) allowing customers to benefit from earlier product releases. A process of continuous planning and feedback ensures that the benefits from the lessons learned are continuously maximized throughout the development process. It also ensures that software teams can continuously align the delivered software with business needs. A key outcome for AGILE is the ability to adapt to changing requirements.

Ultimately, AGILE drives better business outcomes by developing features aligned with customer needs.

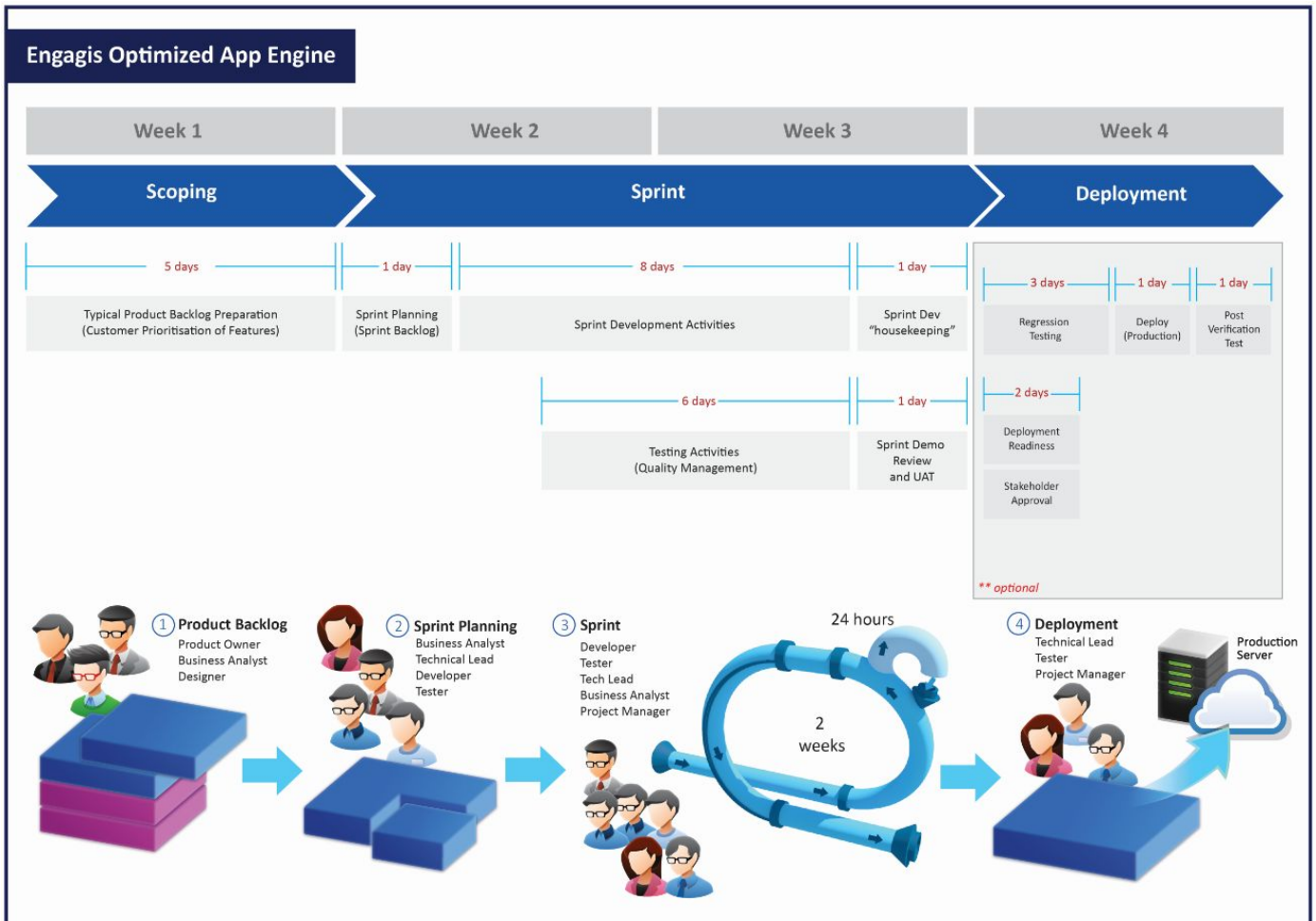


1.2 Benefits of AGILE

Revenue	The iterative nature of agile development means features are delivered incrementally. Benefits are realised early as the product continues to develop.
Speed to Market	At the end of each sprint a potentially shippable build is created. After 80% of the sprints have been completed it may be decided that the project goals have been met and the project can enter the release phases without delivering the remaining features. Projects may run several release phases in between sprints.
Short Feedback loop	A key characteristic of agile development is getting feedback on features as early as possible. This allows additional improvements to be added to the scope if necessary and allows for the quality of the product to be reviewed throughout the development process.
Visibility	AGILE development principles encourage active ‘user’ involvement throughout the product’s development and a very cooperative collaborative approach. This provides excellent visibility for key stakeholders, both of the project’s progress and of the product itself. AGILE helps to ensure that expectations are effectively managed.
Less Risk	Small incremental releases ensures that issues are visible early which makes it easier and faster to respond to change. The clear visibility ensures that decisions can be taken at the earliest possible opportunity, while there’s still time to make a material difference to the outcome.
Response To Change	In traditional development projects, a comprehensive functional specification is written. Changes become complex and expensive to administer. With AGILE development, change is accepted and assumed. And with AGILE, the time-scale is fixed and requirements emerge and evolve as the product is developed. AGILE requires stakeholders who understand this concept and make the necessary trade-off decisions, trading existing scope for new.
Cost Control	The above approach of fixed timescales and evolving requirements enables a fixed budget. The scope of the product and its features are variable, rather than the cost.
Customer Satisfaction	Better business engagement and customer satisfaction is achieved through active involvement of a user representative and/or product owner, the high visibility of the product and progress, and the flexibility to change when change is needed. This is an important benefit that promotes positive and enduring working relationships.
Right Product	With traditional software development processes, the product does not always deliver the benefit that is expected. A short feedback loop allows requirements to be updated in response to feedback during the sprint demo. In AGILE development, the emphasis is on building the right product.
More fun	The active involvement, cooperation and collaboration make AGILE development teams a much more enjoyable place for most people. AGILE creates highly motivated, high performance teams that are extremely cooperative.



2 The Engagis App Engine Process



2.1 Project Scoping

The success of a sprint is directly related to the quality of a project scoping. It consists of capturing customer requirements - determining and documenting a list of specific project goals, deliverables, tasks and deadlines. The purpose of this activity is to define and describe elements within scope of the release in order to clearly understand what will be the area under the project control. Elements not described are considered to be out of scope of the release. Furthermore, the scope of the project provides a view of the number of sprints needed to meet the requirements of the project. The outcome of the scoping session is a Functional Specification document (FSD), a formal Quote and JIRA backlog (all user stories).

2.2 Sprint Explained

Software Development Sprints are defined as a two week period, in which the Engagis Design & Development Team delivers, and makes ready for review, a specific set of work (Sprint Backlog).

Each sprint begins with a planning meeting (Scoping Session), defining the actions and criteria (User Stories) to be met at the completion of the sprint.



During a sprint, the team holds daily stand-up meetings to discuss progress and brainstorm solutions to challenges. This agile approach underpins the success of the Engagis Design & Development Team, by providing Engagis and the client a continual feedback loop, defined delivery times, while reducing up-front client investment.

Team Member	Hours
Head of S&D	4
Senior Project Manager	12
Senior Digital Producer/ Product Owner	18
Senior Tester	40
Senior Developer 1	80
Senior Developer 2	80
Senior Technical Lead	20
Test Lead	38.5
Total	292.5

2.3 Sprint Rules

The following are the ground rules for a sprint:

1. Once the sprint begins, you cannot add requirements or stories to the sprint backlog.
2. User stories should be mature enough to describe the entire scope before they are added to the backlog. The backlog is a set of mature / refined user stories.
3. The user stories do not need to be 100% complete before the sprint is started, but it needs to be detailed enough to describe the entire product scope.
4. The product backlog is created and prioritised with the customer.
5. The sprint duration is 2 weeks (as per the diagram).
6. Team capacity per Sprint is 40 points. One point is an equivalent of one simple login form (full integration with front and backend). This is with a team of 2 x developers, 1 x tester, 25% x of the Business Analyst, 20% x of the Technical Lead.



2.4 Agile Tools

The following are the tools used by Engagis throughout the process:

- **JIRA** – for user stories and acceptance criteria, including Sprint management.
- **Confluence** – documentation and wiki
- **Netsuite** – cases related to deployed solution from the technical support team.



Questions?

1300 203 810

info@engagis.com

www.engagis.com



Sydney • Melbourne • Brisbane • Auckland • Novi Sad • Cagayan de Oro

© 2018 Engagis Pty. Ltd. All rights reserved.

