

Dokumentation IPA yatplaner

IPA in Applikationsentwicklung

Auszubildender - Marius Küng

Auftraggeber - allink.creative GmbH

Projektleiter - Silvan Spross

Chefexperte - Hans Riesenmann

Experte - Antonio Di Luzio

Durchführungsort - allink.creative GmbH

Informatikmittelschule Basel

12.03. - 26.03.2012

Inhaltsverzeichnis

I. Umfeld und Ablauf	3
1. Aufgabenstellung	4
1.1. Titel der Facharbeit	4
1.2. Thematik	4
1.3. Klassierung	4
1.4. Durchführungsblock	4
1.5. Ausgangslage	5
1.6. Detaillierte Aufgabenstellung	5
1.7. Mittel und Methoden	5
1.8. Vorkenntnisse	6
1.9. Vorarbeiten	6
1.10. Neue Lerninhalte	6
1.11. Arbeiten in den letzten 6 Monaten	6
2. Einleitung	7
2.1. Projektorganisation	7
2.2. Vorkenntnisse	7
2.3. Vorarbeiten	8
2.4. Firmenstandards	8
II. Projekt	10
3. Projektbeschreibung	11
3.1. Umfeld	11
3.2. Analyse Wochenplaner (IST-Zustand)	11
3.2.1. Ansichten	11
3.2.2. Funktionsumfang	11
3.3. Definition Muss / Kann Ziele	11
3.4. Lösungsvarianten (SOLL-Möglichkeiten)	11

3.5. Entscheidung Projektleitung Variante	11
4. Realisierung	12
4.1. ERM	12
4.2. Modellierung (Django, aufsetzen Projekt)	12
4.3. Templating Wochenansicht	12
4.4. JS Funktionalität (CRUD mit jQueryUI & AJAX)	12
4.5. Implementierung	12
4.6. Erreichte Ziele	12
5. Testing	13
5.1. Testfälle erfassen	13
5.2. Testfälle ausführen	13
5.3. Testbericht (Ziele)	13
III. Arbeitsjournal	14
5.4. 12.03.2012	15
6. Abbildungsverzeichnis	16
7. Tabellenverzeichnis	17

Teil I.

Umfeld und Ablauf

1. Aufgabenstellung

1.1. Titel der Facharbeit

Webapplikation zur Ressourcenplanung von allink.creative

1.2. Thematik

Es soll eine Webapplikation mit Django erstellt werden, mit welcher die Geschäftsleitung die Ressourcenplanung der Mitarbeiter vornehmen kann. Damit soll eine ältere Webapplikation abgelöst werden.

1.3. Klassierung

- Applikationsentwicklung OO
- UNIX / Linux
- andere Programmiersprache

1.4. Durchführungsblock

Startblock 1: 12.03.2012 - 23.04.2012

IPA-Durchführung: 12.03.2012 - 23.04.2012

Einreichung bis: Montag, 30.01.2012

1.5. Ausgangslage

Bei allink besteht seit Mitte 2010 ein rudimentäres Ressourcenplanungstool. Zur Entwicklung wurden damals jedoch Technologien verwendet, die heute nicht mehr zur Kernkompetenz von allink zählen. Da dieses Tool jedoch jeden Freitag zur Planung der nächsten Woche verwendet wird, ist es seit längerem überfällig es in die bestehende Managementapplikation zu integrieren.

Dank der übermässig langen Testphase des Prototypen sind nun die Anforderungen an das definitive Tool gut bekannt. Daher soll eine Webapplikation mit Django erstellt werden, mit welcher die Geschäftsleitung von allink.creative die Ressourcenplanung der Mitarbeiter vornehmen kann. Damit soll eine ältere Webapplikation abgelöst werden.

1.6. Detaillierte Aufgabenstellung

Das bestehende Tool namens “Yatplaner” soll als Modul im bestehenden Management Tool namens ällink.planer reimplementiert werden. Dabei soll die Bedienbarkeit verbessert werden. Das Ziel ist es das neue Tool so intuitiv bedienen zu können, dass für die Geschäftsleitung keine Schulung nötig ist. Der Praxistest wird voraussichtlich in der letzten IPA Woche an der Wochenplansitzung durchgeführt.

Nebst der begleitenden IPA Dokumentation, wo unter anderem der Funktionsumfang des bestehenden Tools analysiert wird, wird keine zusätzliche Dokumentation gefordert. Der Funktionsumfang des bestehenden Tools soll vom Lernenden in einer Analysephase aufgenommen werden. Dabei sollen die bestehenden Features als Muss- und mögliche neue Features als Kann-Ziele ausformuliert werden.

Der Quellcode des bestehenden Tools ist unter folgender Adresse einsehbar:

<https://github.com/sspross/yatplaner/tree/rails>

1.7. Mittel und Methoden

Folgende Technologien sind zwingend zu verwenden:

- Python 2.6
- Django 1.3
- Piston 2.3
- jQuery 1.8

- HTML5
- CSS3

Das Tool soll in folgenden Browsern fehlerfrei funktionieren:

- Firefox ≥ 8
- Chrome ≥ 10
- Safari ≥ 5

Der Internet Explorer muss explizit nicht unterstützt werden.

1.8. Vorkenntnisse

Dem Lernenden sind alle genannten Technologien bereits bekannt. Seit Beginn seines Praktikums im August 2011 setzt er sich damit auseinander. Gewisse Kombinationen wie z.B. mit jQuery einen AJAX Request zu erstellen, sind jedoch Neuland.

1.9. Vorarbeiten

Es findet keine explizite Vorarbeit statt.

1.10. Neue Lerninhalte

Wie bereits erwähnt sind dem Lernenden alle Technologien bereits bekannt. Jedoch sind gewisse Kombinationen noch nie vom Lernenden selbst angewandt worden. Das Know-how ist bei allink ausreichend vorhanden. Der Lernende kann zudem auf eine Vielzahl von bestehenden Projekten zurückgreifen, wo er unzählige Beispiele studieren kann.

1.11. Arbeiten in den letzten 6 Monaten

Der Lernende hat überwiegend Webseiten mit den oben genannten Technologien umgesetzt. Zu seinen umfassendsten Arbeiten zählen bis jetzt eine Webseite eines Immobilienunternehmens und einer Eventplattform eines Finanzkonzerns. Für ersteres arbeitete der Lernende rund 250 Stunden daran.

2. Einleitung

2.1. Projekorganisation

Projekorganisation

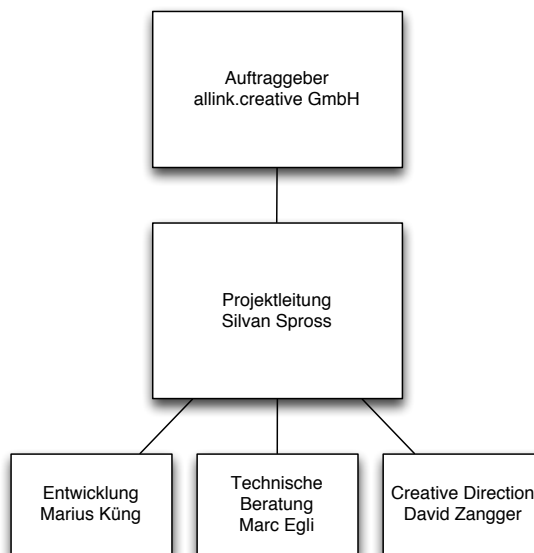


Abbildung 2.1.: Projekorganisation¹

2.2. Vorkenntnisse

Technologien

- Python Grundkenntnisse
- Django Fortgeschrittene Kenntnisse

¹Eigene Darstellung

- Piston Grundkenntnisse
- jQuery Fortgeschrittene Kenntnisse
- HTML5 Gute Kenntnisse
- CSS3 Gute Kenntnisse
- AJAX Fortgeschrittene Kenntnisse

Anwendungen

- Mehrere komplette Webauftritte realisiert
- Applikationen in Django erstellt (News, Blog, Produkteübersicht)
- Schnittstellen programmiert (XML, JSON)
- Per AJAX dynamische Inhalte laden und einfügen
- Dynamische Formulare abschicken und Request entgegennehmen
- DOM-Elemente manipulieren, per Events steuern

2.3. Vorarbeiten

Ich habe, um einen ersten Eindruck über die Funktionalität zu erhalten, den bestehenden yatplaner ausprobiert. Ausserdem den allink.planer studiert in den, der Wochenplaner implementiert wird. Ansonsten fand keine explizite Vorarbeit statt.

2.4. Firmenstandards

- Betriebssystem: Mac OS X
- Editor: Textmate
- Entwicklungsumgebung: Python, Django (Python-Framework für Webapplikationen)
- Virtuelle Testumgebung: Django Serversimulation (per Terminal steuerbar)
- Deployment: per fabric-script auf Apache-Server mit WSGI-Protokoll
- Versionierung: Git, auf Github gehostet

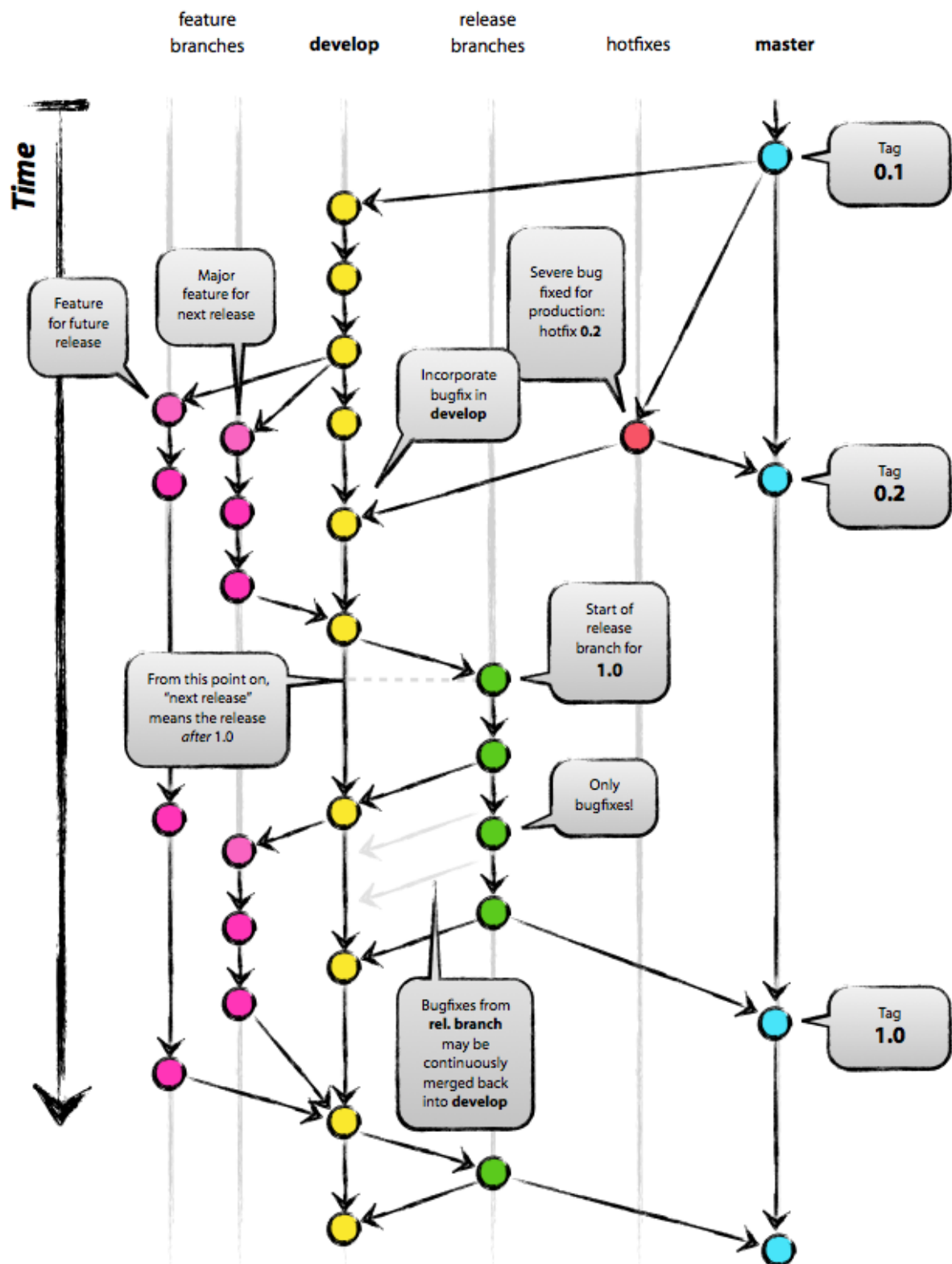


Abbildung 2.2.:
"A successful Git branching model" von Vincent Driessen

Teil II.

Projekt

3. Projektbeschreibung

3.1. Umfeld

Ist dieser Abschnitt nötig?

3.2. Analyse Wochenplaner (IST-Zustand)

3.2.1. Ansichten

3.2.2. Funktionsumfang

3.3. Definition Muss / Kann Ziele

3.4. Lösungsvarianten (SOLL-Möglichkeiten)

3.5. Entscheidung Projektleitung Variante

4. Realisierung

4.1. ERM

4.2. Modellierung (Django, aufsetzen Projekt)

4.3. Templating Wochenansicht

4.4. JS Funktionalität (CRUD mit jQueryUI & AJAX)

4.5. Implementierung

4.6. Erreichte Ziele

5. Testing

5.1. Testfälle erfassen

5.2. Testfälle ausführen

5.3. Testbericht (Ziele)

Teil III.

Arbeitsjournal

5.4. 12.03.2012

Beginn IPA yatplaner

- Zeitplan erstellen (Planung)
- Aufgabenstellung erfassen
- Projektorganisation grafisch erfassen
- Designbesprechung mit Dave
- Vorkenntnisse erfassen
- Vorarbeiten erfassen
- Firmenstandards erfassen

6. Abbildungsverzeichnis

2.1. Projekorganisation	7
2.2. “A successful Git branching model” von Vincent Driessen	9

7. Tabellenverzeichnis