



Abschlussprüfung Winter 2018

Fachinformatikerin für Anwendungsentwicklung

Dokumentation zur betrieblichen Projektarbeit

Datum: 12.11.2018

SLA (Service Level Agreement) Reporting der internen „issue tracking and project management platform,, YouTrack

Prüfungsbewerber

Yen Nguyen

Freibadstraße 1

81543 München



Ausbildungsbetrieb

iXenso AG

OpenText Platinum Partner

Munzinger Str. 1

79111 Freiburg

Inhaltsverzeichnis

Inhaltsverzeichnis.....	I
Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
Abkürzungsverzeichnis	VI
Listingsverzeichnis.....	VII
1 Einleitung.....	1
1.1 Projektbeschreibung	1
1.2 Projektziel.....	1
1.3 Projektumfeld	2
1.4 Projektbegründung	2
1.5 Projektschnittstellen.....	3
1.6 Projektabgrenzung	3
2 Projektplanung.....	4
2.1 Projektphasen	4
2.2 Ressourcenplanung	4
2.3 Entwicklungsprozess	4
3 Analysephase.....	6
3.1 Ist-Analyse	6
3.2 Soll-Konzept	6
3.3 Erstellung des Lastenheftes.....	7
3.4 Wirtschaftlichkeitsanalyse	7
4 Entwurfsphase	9
4.1 Zielplattform	9
4.2 Architekturdesign.....	9

4.3 Erstellen von Benutzeroberfläche (Mock-Ups).....	10
5 Implementierungsphase.....	11
5.1 Implementierung der Benutzeroberfläche.....	11
5.2 Geschäftslogik implementieren.....	11
6 Qualitätsmanagement.....	13
6.1 Durchführung von Tests.....	13
6.2 Fehlerbehebung/Korrekturen	13
7 Abnahme- und Deployment.....	13
7.1 Abnahme durch Projektleiter	13
7.2 Deployment der Anwendung	14
8 Erstellen der Dokumentation	15
8.1 Projektdokumentation.....	15
8.2 Anwenderdokumentation	15
9 Projektbewertung.....	16
9.1 Soll-/Ist-Vergleich.....	16
9.2 Fazit	16
9.3 Ausblick	16
10 Literaturverzeichnis.....	17
A. Anhang	i
A.1. Detaillierte Zeitplanung.....	i
A.2. Reale Dauer der Projektphasen in Tortendiagramm ..	ii
A.3. Gantt Diagramm	iii
A.4. Verwendete Ressourcen	iv
A.5. Use-Case-Diagramm	v
A.6. Lastenheft (Auszug).....	vi
A.7. Oberflächenentwürfe (Mock-Ups)	vii
A.8. Pflichtenheft(Auszug).....	viii

A.9. Komponentendiagramm	ix
A.10. Screenshots der Start Seite	x
A.11. Screenshots der Tabellenseite.....	xi
A.12. Listings der Cookie Authorization	xii
A.13. Listing – Class Controller	xiii
A.14. Listing - Class Bavaria_Holidays.....	xiv
A.15. Listing – Index.php	xv
A.16. Listings – Declaration und Navigation der HLML Seite.....	xvi
A.17. Listing – Methode „excel_export“ in Class PHPExcel_Download	xvii
A.18. Listing – Class Business_Datetime	xviii
A.19. Listings – Event-Handler der API-Schnittstelle (Auszug von start_form.js)	xix
A.20. Listings - Auszug von Klasse Home_Controllerxx	
A.21. Listing – Class Ticket_Model	xxi
A.22. Listing – Class Project_Model	xxii
A.23. Anwenderdokumentation	xxiii
A.24. Get_business_time_duration Beschreibung .	xxiv
A.25. Klassendiagramm	xxv

Abbildungsverzeichnis

Abbildung 1: Projectphasen in Tortendiagrammii

Abbildung 2: Gantt Diagramm iii

Abbildung 3: Use-Case-Diagramm v

Abbildung 4: Mock-Up der Start Seite vii

Abbildung 5: Mock-Up der Tabelle Seite vii

Abbildung 6: Komponentendiagramm..... ix

Abbildung 7: Screenshot der Start Seite x

Abbildung 8: Datum Felder mit Standard Wert und Datepicker x

Abbildung 9: Screenshot der Tabellenseite xi

Abbildung 10: exportierte Excel Report Datei xxiii

Abbildung 11: Klassendiagramm xxv

Tabellenverzeichnis

Tabelle 1: Aufstellung der Excel Tabelle2

Tabelle 2: Grobe Zeitplanung.....4

Tabelle 3: Kostenaufstellung8

Abkürzungsverzeichnis

API.....Application Programming Interface

CASE – Tools..... *Computer-aided software engineering Tools, Computer-aided software engineering Tools*

ECM.....*Enterprise Content Management*

PMO*Project Management Office*

SLA *Service Level Agreement*

Listingsverzeichnis

Listing 1: Cookie Authorization..... xii

Listing 2: Class Controller..... xiii

Listing 3: Klasse Bavaria_Holidays xiv

Listing 4: File index.php xv

Listing 5: Declaration und Navigation der HLML Seite xvi

Listing 6: Methode "excel_export" xvii

Listing 7: Class Business_Datetime..... xviii

Listing 8: Event-Handler JavaScript der API Schnittstelle..... xix

Listing 9: Klasse Home_Controller xx

Listing 10: Klasse Ticket_Model xxi

Listing 11: Klasse Projekt_Model..... xxii

1 Einleitung

Die folgende Projektdokumentation schildert den Ablauf des IHK-Abschlussprojektes, welches die Autorin im Rahmen ihrer Ausbildung zur Fachinformatikerin für Anwendungsentwicklung durchgeführt hat.

Der zuständige Ausbildungsbetrieb ist die iXenso AG München, ein Open Text Platinum Partner für ECM Consulting Partner & Reseller für Enterprise Content Management (ECM).

1.1 Projektbeschreibung

Der Auftraggeber des Projekts ist die iXenso AG München. Ansprechpartner für die Anforderungen an die Umsetzung ist Herr Matthias van Hout vom iXenso Support Team.

Eine interne Reporting Webanwendung wird entwickelt, um über die API Schnittstelle Daten der Issue Tracking and Project Management Plattform (YouTrack) auszulesen. Die Werte sollen nach Projekt und Zeitraum gefiltert, angezeigt, sortiert und nach Excel exportiert werden.

1.2 Projektziel

Ziel des Projektes ist die Entwicklung einer SLA Reporting Webanwendung, um das bestehende Ticketsystem zu analysieren. Die Anwendung soll das Auslesen der Daten aus der Benutzeroberfläche des tracker.ixenso.com (YouTrack) automatisieren.

Diese Anwendung soll in PHP mit dem nötigen Framework und Programmcode realisiert werden.

Die Excel Datei soll die folgenden Informationen enthalten:

- Darstellung der Informationen des Projektes
 - Name des Projekts
 - SLA Zeiten (in der ersten Version werktags, Mo-Fr 08.00 Uhr bis 17.00 Uhr, außer WE und Feiertage, nach Bundesländer und Länder, die Zeitzone berücksichtigen)
 - Gültiger Feiertagskalender
- Mit Informationen von:
 - **Name des Tickets** (ticket): Liste aller Ticketnamen eines Projektes
 - **Erstellt von** (created by): Wer hat das Ticket erstellt
 - **Erstellzeitpunkt** (create date): Wann wurde das Ticket erstellt
 - **Reaktionszeit** (time to respond): Die Reaktionszeit beginnt mit der Erstellung des Tickets in YouTrack bis zur erste Reaktion eines Bearbeiters.

- **Lösungszeit** (time to resolved): Von der Erstellung bis Status (State) „closed“ oder „verified“, je nachdem was als erstes passiert.
- **Aktueller Status** (current status): Aktueller Bearbeitungsstatus des Tickets
- **Priorität** (priority): Priorität des Tickets

Ticket	Created by	Create Date	Time to respond	Time to resolve	Current status	Priority

Tabelle 1: Aufstellung der Excel Tabelle

1.3 Projektumfeld

Die iXenso AG ist ein OpenText Platinum Partner & Reseller für Enterprise Information Management und implementiert, auf Basis der OpenText Suiten (ECM, BPM, CEM, IX) seit 1999 Software-Lösungen zur Optimierung von Geschäftsprozessen - insbesondere im Umfeld von SAP R/3. Das inhabergeführte und global tätige Beratungs- und Dienstleistungsunternehmen hat sich auf Capture & Delivery, BPM, DMS, Webportale, mobile Anwendungen sowie die revisionssichere Archivierung spezialisiert und wurde von OpenText bereits mehrfach ausgezeichnet, u.a. als 'EMEA PARTNER OF THE YEAR 2014' und 'BEST SOLUTION PARTNER 2013'.

Die Anforderungen für dieses Projekt stammen von Herrn Udo Kemle, dem Projektleiter (PMO) der iXenso AG, sowie Herrn Matthias van Hout vom iXenso Support Team.

Das System soll komplett web-basiert konzipiert sein, so dass dessen Oberflächen unter Verwendung von PHP entwickelt werden kann.

1.4 Projektbegründung

Momentan beinhaltet die Problemverfolgung und das Projektmanagement mittels YouTrack hat ein hohes Maß an manueller Arbeit. Den elektronischen Datenverarbeitungsfluss von der Erstellung eines Tickets bis zur Fertigstellung händisch zu verfolgen ist ziemlich aufwendig. Hierbei kann es schnell zu Flüchtigkeitsfehlern kommen, die dann wiederum zu Folgefehlern führen. Außerdem könnte es sogar passieren, dass einzelne Tickets komplett vergessen werden.

Ein weiteres Problem ist, dass die eingebauten Reports, in welchen die Daten in Grafiken dargestellt werden, durch die stetig wachsende Größe ihre Übersichtlichkeit verlieren. Zudem ist diese nicht ausreichend, da nur die

allgemeine Information angezeigt werden. Andere Information und Berechnungen, wie z.B. die Reaktionszeit (time to respond), Lösungszeit (time to resolved), werden in diesen Berichten nicht mit angeboten. Dies hat den Nachteil, dass eine entsprechende Analyse noch einmal extra erstellt werden muss.

Aufgrund dieser Probleme hat sich die iXenso AG München dazu entschieden, die Entwicklung einer Webanwendung in Auftrag zu geben, durch die der gesamte Berichterstattungsprozess automatisiert werden soll. Dies ist unerlässlich in einer schnelllebigen und vernetzten Arbeitswelt.

1.5 Projektschnittstellen

Damit die für den Verarbeitungsprozess relevanten Daten vollständig ermittelt und validiert werden können sowie der Status des Tickets automatisch gelesen werden kann, muss die Webanwendung mit YouTrack interagieren können. Diese Anforderung soll mit Hilfe von REST API, welche eine leistungsstarke Schnittstelle darstellt, umgesetzt werden. Bei der REST API handelt es sich um eine Client Bibliothek, welche in diesem Fall den Export der verschiedenen Projekte durchführt und die dafür benötigten Funktionen zur Verfügung stellt. Das Erstellen der Datentabellen und Exportieren nach Excel erfolgt direkt über den Aufruf der SLA Reporting Anwendung. Aus diesem Grund muss eine PHP Klasse implementiert werden, um automatisch Vorgänge für die Berechnung der Übertragungswerte zu erzeugen.

1.6 Projektabgrenzung

Es handelt sich um ein komplett eigenständiges Projekt ohne Bezug zu anderen Projekten/Teilprojekten. Die Erarbeitung inhaltlicher Fragestellungen waren nicht Teil des Projektes.

2 Projektplanung

2.1 Projektphasen

Für die Umsetzung des Projektes standen 70 Stunden zur Verfügung. Diese wurden vor Projektbeginn auf verschiedene Phasen verteilt, die während der Softwareentwicklung durchlaufen werden. Eine grobe Zeitplanung sowie die Hauptphasen lassen sich der folgenden Tabelle entnehmen:

	Schritt	Dauer
1	Analysephase	11
2	Entwurfsphase	6
3	Implementierungsphase	32
4	Qualitätsmanagement	7
5	Abnahme- und Deployment-Phase	5
6	Erstellen der Dokumentation	6
7	Projektbewertung	3
	Gesamt	70h

Tabelle 2: Grobe Zeitplanung

Zusätzlich können die einzelnen Hauptphasen in Unterpunkte aufgeteilt werden. Eine detaillierte Übersicht dieser Phasen befindet sich im Anhang A.1: Detaillierte Zeitplanung auf S. i.

2.2 Ressourcenplanung

Eine der im Zusammenhang mit der Einführung vom Projekt am häufigsten diskutierten Aspekte betrifft die Wirtschaftlichkeit. Dabei geht es einerseits darum, die Bindung von Ressourcen möglichst gering zu halten und andererseits darum, über das Ergebnis eines Prozesses möglichst frühzeitig verfügen zu können. In der Übersicht, welche sich im Anhang A.2: Verwendete Ressourcen auf S. ii befindet, sind alle Ressourcen aufgelistet, welche für dieses Projekt eingesetzt wurden. Damit sind sowohl Hard- und Softwareressourcen als auch das Personal gemeint.

Bei der Auswahl der verwendeten Software wurde darauf geachtet, dass diese kostenfrei (z.B. als Open Source) zur Verfügung steht oder die iXenso AG bereits Lizenzen für diese besitzt.

2.3 Entwicklungsprozess

Entwicklungsprozesse beschreiben die Vorgehensweise bei der Entwicklung von neuer Software. Das Ziel ist es, einen idealen Entwicklungsprozess zu schaffen, der mit möglichst geringen Kosten bei optimalem Ergebnis auskommt. Zur Kostenreduktion eignet sich beispielsweise die Verbesserung der Kommunikation

durch die Verwendung von CASE – Tools oder die Vermeidung von Arbeitsschleifen in dem Prozess.

Der Realisierung des Projekts vorgehend ist die Entscheidung des Autors bezüglich eines Entwicklungsprozesses. An dem ausgewählten Entwicklungsprozess orientiert sich die gesamte Vorgehensweise der Umsetzung. Hierbei gilt es die Anforderungen an die Umsetzung zu beachten. Da im Rahmen dieses Projekts mehrere einzelne Teilergebnisse voneinander unabhängig angefertigt werden, wurde hier ein agiler Entwicklungsprozess ausgewählt. Dieser erlaubt ein iteratives Durchlaufen der Projektphasen und ermöglicht zudem kontinuierliche Rücksprache mit den Stakeholdern.

3 Analysephase

Nachdem die Planung des Projektes abgeschlossen ist, kann mit der Analysephase zur Ermittlung des Ist-Zustandes begonnen werden.

3.1 Ist-Analyse

YouTrack ist ein proprietärer, kommerzieller browserbasierter Bug Tracker (ein Issue-Tracking-System bzw. Projektmanagement-Software), welche von dem Hersteller JetBrains entwickelt wurde. Es konzentriert sich auf haben die Möglichkeiten der Zuweisung eines Tickets an eine Funktionsstelle oder an eine Person innerhalb einer Funktionsstelle zur weiteren Bearbeitung bis zur Lösung. Das Issue Tracking und Projektmanagement Tool YouTrack erfüllt verschiedene Funktionen, insbesondere:

- (1) YouTrack ist für Entwickler und agile Teams optimiert. Es beinhaltet die Issues (d.h. Erstellung der Einträge bis zur Fertigstellung und verfolgt alles, woran das Team arbeitet)
- (2) Projekte organisieren und einrichten, dass die Störungen, Fehlern und Anfragen erfasst werden.
- (3) Verteilung und Zuordnung der Bearbeiter. Überwachung der Bearbeitung und der Bearbeitungsdauer und -qualität
- (4) Garantieren des Einhaltens interner Abläufe durch Zwangssteuerung über Workflows
- (5) Zeiterfassung verwenden, um die aufgewendete Zeit für verschiedene Projekte und Arten von Aktivitäten zu überwachen und zu melden.
- (6) Die Produktentwicklung auf einem agilen Board planen und priorisieren.
- (7) Berichte erstellen, mit denen man Informationen sammeln und zusammenfassen kann.
- (8) Ein Dashboard erstellen, um die Aktivitäten in allen Projekten zu überwachen.

Wie bereits im Abschnitt „1.4 Projektbegründung“ beschrieben, wird YouTrack bisher über eine webbasierte Management-Oberfläche manuell verwaltet. Bislang ist es nur möglich, über diese Oberfläche Projektdaten einzusehen. Es gibt kein SLA Reporting zur eingehenden Analyse der Projekt- und Ticketdaten.

Die Umsetzung dieses Projektes ist aufgrund der Anforderung eines SLA Reportings aus YouTrack notwendig.

3.2 Soll-Konzept

Es gilt jetzt auszuarbeiten, wie ein Gesamtbericht aus dem Ticketsystem ausgelesen und exportiert werden kann. Im Vordergrund steht hier natürlich das Ziel, dem Mitarbeiter den Bearbeitungsaufwand, der in Abschnitt 3.1 Ist-Analyse beschrieben ist, zu minimieren. Vorab sollen alle Projektnamen mit allen

dazugehörenden Tickets ausgelesen werden. Aus einem ausgewählten Projekt soll ein einstellbarer Zeitraum berücksichtigt werden, zu dem eine Auflistung von Tickets erfolgt. Hiermit soll der SLA Report sofort heruntergeladen werden können oder die relevanten Report-Daten zu diesem Projekt in Tabellen angezeigt werden. Hier soll zusätzlich eine Sortierung möglich sein. Die Anforderungen werden noch im Zuge der Erstellung des Lastenheftes, siehe Abschnitt 3.3 „Erstellung des Lastenheftes“, in Zusammenarbeit mit den Stakeholdern genauer definiert.

3.3 Erstellung des Lastenheftes

Für die Gewährleistung der Vollständigkeit, entsprechend der Anforderungen der Stakeholder, mussten diese ein Lastenheft erstellen. Hierbei hat die Autorin geholfen, so dass die Anforderungen vollumfänglich vorhanden und klar definiert sind. Ein Auszug aus diesem Lastenheft, welcher einen Teil der Anforderungen darstellt, befindet sich im Anhang A.4 Lastenheft (Auszug).

3.4 Wirtschaftlichkeitsanalyse

Durch eine Betrachtung der Projektkosten soll in diesem Kapitel untersucht werden, ob das Projekt auch wirtschaftlich gerechtfertigt ist.

3.4.1 „Make or Buy“-Entscheidung

Aufgrund der Individualität kommen keine Branchenlösungen in Frage. Auch eine externe Entwicklung kommt aufgrund eines hohen Preises für die Umsetzung der gewünschten Funktionalitäten nicht in Frage. Daher wird die Entscheidung getroffen, die Anwendung intern zu entwickeln.

3.4.2 Projektkosten

Im Folgenden werden die bei der Projektdurchführung entstandenen Kosten kalkuliert. Die Projektkosten, die während der Entwicklung des Projektes anfallen, sollen im Folgenden kalkuliert werden. Dafür müssen neben den Personalkosten, die durch die Realisierung des Projektes verursacht werden, auch noch die Aufwendungen für die Ressourcen (Hard- und Software, Büroarbeitsplatz sowie benötigtes Personal etc.) berücksichtigt werden. Da die genauen Personalkosten nicht veröffentlicht werden dürfen, wird die Kalkulation mit Stundensätzen durchgeführt, die von der Personalabteilung vorgegeben werden. Der Stundensatz eines Auszubildenden beträgt 15,- EUR, der eines Angestellten 85,- EUR und für die Projektabnahme 90,- EUR.

Die Kosten, welche für die einzelnen Vorgänge des Projektes anfallen sowie die gesamten Projektkosten lassen sich der Tabelle 3: Kostenaufstellung entnehmen:

Vorgang	Zeit (Stunde)	Ressourcen	Gesamt
Entwicklungskosten	57	15,00 €	855,00 €
Projektmanagement	4	85,00 €	340,00 €
Qualitätsmanagement	7	85,00 €	595,00 €
Abnahme	2	90,00 €	180,00 €
Projektkosten Gesamt			1.970,00 €

Tabelle 3: Kostenaufstellung

4 Entwurfsphase

Als Folge der Analysephase wurde vor der eigentlichen Implementierung des Projektes der technische Entwurf durchgeführt. Hierbei wird dargestellt, wie das System später aussehen soll und wie dies technisch umzusetzen ist. Am Ende der Entwurfsphase entsteht das Pflichtenheft, welches den Auftraggebern des Projektes vorgelegt wird.

4.1 Zielplattform

Die Entwicklung dieses Projektes erfolgt in deutscher Sprache, dies ist durch die Architekturrichtlinien der iXenso AG so vorgegeben. Außerdem soll dieses Projekt als Webanwendung umgesetzt werden. Die Anmeldung zur Anwendung ist nicht erforderlich, da das Programm nur Daten lesen und über einen Link im Webbrowser aufgerufen werden soll.

Die Anwendung wird in der Programmiersprache PHP programmiert. PHP ist eine Skriptsprache, welche hauptsächlich im Webbereich zum Einsatz kommt, um dynamische Dokumente zu erstellen, wodurch eine einfache Wartung des Programmcodes gewährleistet ist. Für diese Webanwendung wird die Spezifikation PHP 7.2 verwendet. Diese Spezifikation ist die neueste Technologie, welche im Internet veröffentlicht ist. Die Webanwendung wird auf der internen Application Platform ausgeführt. Ein Applicationserver ist ein Server innerhalb eines Computernetzwerks, der Anwendungsprogramme ausführt.

4.2 Architekturdesign

Das Projekt basiert auf dem Architekturmuster Model View Controller (MVC). Das Architekturmuster ist in einem Komponentendiagramm im Anhang A.8, Seite ix dargestellt.

Gemäß diesem Muster lässt sich die Software in die drei Einheiten Model (Datenhaltung), View (Präsentation) und Controller (Anwendungssteuerung) unterteilen. Im Model werden die Projektdaten und deren Ticketdaten mit ihren Eigenschaften und Methoden gespeichert und verwaltet. Ein Projekt Model ist ein Konstrukt, in dem Informationen eines Projekts abgespeichert und bearbeitet werden können. Ein Projekt Model enthält mehrere Ticket Modelle. Der Controller ist als Verbindung zwischen dem Model und der View für die Steuerung der Daten zuständig. Im View werden Daten abgebildet und Benutzereingaben entgegengenommen. Die JetBrains Version YouTrack REST API PHP Client von Anton Komarev, version v6.1.0 ist für den Zugriff auf die Schnittstelle der

YouTrack zuständig. Der Controller mit einem Objekt der Client von REST API liest die Daten von der Tracking System YouTrack, welche die Modelle verarbeiten. Diese Daten werden mit PHPEXcel in Excel Dateien exportiert. Die View Anzeige auf der View Template übergibt die Daten über die Benutzereingabe an den Controller, um diese weiterzubearbeiten. Die Ausgabe erfolgt über einen Bericht mit allen erforderlichen Daten.

4.3 Erstellen von Benutzeroberfläche (Mock-Ups)

Um die Anwendung möglichst benutzerfreundlich bedienen zu können, soll eine klar strukturierte, einfache Benutzeroberfläche entwickelt werden. Mit Hilfe von Mockups wurde hierfür zunächst ein Prototyp der Oberfläche angefertigt. Dafür wurde Bootstrap und JQuery UI verwendet. Damit die Benutzeroberfläche am Ende den Anforderungen und Vorstellungen des Fachbereichs entspricht, wurden diese bei der Entwurfsphase intensiv mit einbezogen. In der Hauptansicht sollen zu sehen sein: Liste von Projektnamen, Inputfelder von Datum und Uhrzeit, Buttons. Die Mockups der Hauptansicht sowie der Tabellenseite für den Berichtsexport befinden sich im Anhang A.5: Oberflächenentwürfe (Mock-Ups) auf S. vii.

5 Implementierungsphase

Zur Entwicklung des Frontend nutzte die Autorin die PHP Entwicklungsumgebung für Eclipse Plattform.

5.1 Implementierung der Benutzeroberfläche

Da hier das Prinzip von MVC Anwendung finden soll, hat sich die Autorin dafür entschieden mit der Erstellung der Benutzeroberflächen anzufangen. Hier wurde mit Hilfe von Bootstrap sowie zusätzlichen Elementen von JQuery, JQuery UI¹ und Fontawesome² entwickelt. Die Funktionalität der einzelnen Bestandteile der GUI wurde im Code-Behind definiert. Screenshots der fertigen Oberflächen befinden sich im Anhang A.10 und A.11.

5.2 Geschäftslogik implementieren

Die Implementierung der Geschäftslogik stellt den größten Entwicklungsaufwand dar und muss möglichst effizient durchgeführt werden. Zuerst mussten die entsprechenden Klassen gemäß Anhang A.25 Klassendiagramm erstellt werden. Zwei Beispiele befinden sich im Anhang A.13 und A.14 Listing Klasse Bavaria_Holidays und Controller. Eine Login Maske ist für den User Login aufgebaut. Der Username und das Passwort müssen mit dem YouTrack Account übereinstimmen. Die Funktion `get_holidays_array($year)` in Klasse Bararia_Holidays werden die Feiertage im ausgewählten Jahr als Parameter zurückliefern. Jede Controller Klasse soll eine Eigenschaft von einem Client Objekt der YouTrack API haben.

Im Anschluss daran wurde die Controller Klasse für die Startseite und Tabellenseite erstellt, so dass jede Klasse die Aktionen als Methode von jeder Seite erhalten. Es gibt 2 Modelle je Klasse: *Project_Model* und *Ticket_Model*. Die Eigenschaft *ticket_array* kommt vom *Project_Model* Objekt, die *Ticketdaten* als Objekt sind vom Projekt *Ticket_Model*. Die Ergebnisse der noch zu erstellenden Methoden „*get_projects*“ und „*get_tickets*“ werden direkt in einem Array gespeichert und angezeigt. Der Aufbau der neuen Klassen *Project_Model* und *Ticket_Model* befindet sich im Anhang A.21 und A.22.

Nachdem die Modellklasse erstellt wurde, wurde die Controller Methode definiert. Hierbei wurden mittels der Methode *get_projects* alle Projektnamen ausgelesen.

¹ Vgl. <https://jqueryui.com/>

² Vgl. <https://fontawesome.com/>

Die Implementierung der Methode befindet sich im Anhang A.19 Methoden der Rest API. In dieser Methode wurde vorerst die Methode `get()` der Client Object aufgerufen, um die Ticketdaten von YouTrack zu holen und danach in Select Option Liste darzustellen. *From* und *To* Datum werden als Standard Wert von den letzten 3 Monaten übernommen, falls keine neuen Werte ausgewählt wurden. Die Service time *From* und *To* werden mit Standard Werte von 08:00 Uhr bis 17:00 Uhr definiert, jeder Tag enthält eine Stunde Pause sein, sonst werden die Werte von Eingabe der User übernommen.

Der nächste Schritt ist die Implementierung der Methode `get_ticket`. Hier werden alle Ticketdaten eines Projekts bearbeitet. Die Parameter der Methode `get_tickets` werden definiert von einem Array der Informationen wie Projectname, von Datum und bis Datum sowie der Supportzeit. Durch ein JavaScript Event-Handler von Button „Show Report“, wird die `get_tickets` Methode aufgerufen. Nachdem man die aktuellen Ticketdaten über diese Eigenschaft gespeichert hat, wird die Reaktionszeit (*time to respond*) und die Lösungszeit (*time to resolved*) in der Methode `get_business_time_duration` kalkuliert. In dieser Methode werden die Feiertage sowie die verfügbare Arbeitszeit berücksichtigt. Die Feiertage in Bayern werden von einer Methode der Klasse `Bavaria_Holidays` aufgenommen. Eine Beschreibung zu dieser Methode ist im Anhang A.24 `get_business_time_duration` Beschreibung zu finden.

Es wird nun auf die View Template zurückgegriffen, welche letztendlich eine Tabelle mit den Ergebnissen der Ticketdaten anzeigt: das ausgewählte Projekt und die Anzeige des Zeitraumes. Zusätzlich kann man hier die Ticketdaten sortieren und in eine Exceldatei exportieren.

Im Anhang A.19 Event-Handler der API-Schnittstelle befindet sich ein Auszug der entwickelten JavaScript Event-Handler. Zusätzlich zu den dort aufgezeigten Event-Handlern wurden noch einige Error Event-Handler implementiert: Error anzeigen, wenn kein Projekt ausgewählt wurde oder keine Tickets vom ausgewählten Projekt bzw. kein Zeitraum gefunden wurde.

Wenn der Button „Download“ angeklickt wurde, werden die Methode der Klasse `PHPExcel_Download` aufgerufen, um die Exceldatei zu erstellen und herunterzuladen. Im Anhang A.17 Listing-Methode `excel_export` befindet sich ein Auszug über die im Rahmen der Exceldatei exportierten und implementierten Methoden.

Man kann auch direkt den SLA Report als Exceldatei herunterladen über die Funktion Start Form mit dem Button „Download“.

6 Qualitätsmanagement

6.1 Durchführung von Tests

Die Testdurchführung ist eine Aktivität zur Bewertung der Produktqualität des Projects und zum Aufspüren noch vorhandener Fehler. Das Testen bezieht sich hiermit auf folgende Fragestellungen: Der Integrationstest betrachtet das Zusammenspiel (wie Klassen oder Module). Ein Systemtest bewertet das Verhalten der gesamten Software.

Die Reaktionszeit und das Verhalten werden ebenfalls getestet sowie der fachliche Akzeptanztest durch die Anwender. Dieser erfolgt mittels eines Abnahmetestes durch nachfolgende Auftraggeber: Herrn Matthias van Hout und Herrn Udo Kemle.

Beim Testen werden für vorab definierte Eingabewerte die erwarteten Werte mit den tatsächlichen Ergebniswerten verglichen. Es werden Black-Box und White-Box Test durchgeführt. Beim Black-Box Test wird die SLA Report Anwendung nur anhand seiner Spezifikation (Projektname auflisten, Tickets Tabelle und Excel exportieren) getestet. Der White-Box-Test ist besonders beim Komponententest wichtig. Beim White-Box Test wird der innere Aufbau eines Programms mitberücksichtigt.

6.2 Fehlerbehebung/Korrekturen

Die objektorientierte Programmierung besitzt gegenüber der klassischen imperativen Programmierung einige strukturelle und dynamische Besonderheiten (Kapselung, Polymorphie, Vererbung), welche bei der Konzeption von Tests zu berücksichtigen sind.

7 Abnahme- und Deployment

7.1 Abnahme durch Projektleiter

Nachdem die gesamte Anwendung fertiggestellt wurde, konnte diese dem Fachbereich zur Endabnahme vorgelegt werden. Aufgrund der agilen Softwareentwicklungsmethode wurde den Fachbereichen nach jeder Iteration die aktuelle Version der Anwendung präsentiert. Dadurch waren sie bei der Endabnahme bereits mit der Oberfläche und der Funktionsweise des Programmes vertraut. Außerdem konnten Anregungen und Kritik der Fachbereiche durch die stetigen Rücksprachen schon frühzeitig während der

Entwicklungsphase berücksichtigt werden. Dadurch ergaben sich bei der Endabnahme keine Probleme oder Hindernisse mehr, sodass der Einführung der Anwendung nichts mehr im Wege stand. Vor dem Deployment und der Qualitätssicherung wurde zusätzlich zur Abnahme durch den Fachbereich ein Code-Review durch einen anderen Entwickler durchgeführt.

7.2 Deployment der Anwendung

Um den Zugriff auf die Anwendung für die Fachbereiche möglichst einfach zu gestalten, wurde die Anwendung zentral über ein bereits existierendes Netzlaufwerk (für das alle Beteiligten die nötigen Berechtigungen haben), bereitgestellt. Die Anwendung wurde als kompiliertes, ausführbares Programm auf diesem Laufwerk abgelegt. Außerdem wurde bei jedem Benutzer eine Desktopverknüpfung angelegt, um einen schnellen und einfachen Zugriff auf die Anwendung zu gewährleisten. Wie bereits im vorherigen Abschnitt erwähnt wurde, waren die Fachbereiche durch die stetige Rückkopplung schon mit der Anwendung vertraut. Daher waren zusätzliche Benutzerschulungen nicht erforderlich.

8 Erstellen der Dokumentation

Die gesamte Dokumentation setzt sich aus zwei Bestandteilen zusammen: der Projektdokumentation und der Anwenderdokumentation.

8.1 Projektdokumentation

In der Projektdokumentation beschreibt die Autorin die gesamte Planung und Umsetzung des Projekts, die während der Projektumsetzung durchlaufen wurden. Sie wurde während der gesamten Bearbeitungszeit angefertigt.

8.2 Anwenderdokumentation

Die Anwenderdokumentation soll den Benutzern des Moduls einen Überblick über die Funktionalität bieten und ein Anhaltspunkt sein. Ein Auszug daraus befindet sich im Anhang A.23 Anwenderdokumentation(Auszug).

9 Projektbewertung

9.1 Soll-/Ist-Vergleich

Die Gesamtzeit von 70 Stunden wurde eingehalten, ein paar Punkte wiesen jedoch Abweichungen vom geplanten Zeitaufwand auf. Dazu zählen unter anderem die Punkte 6.2 Fehlerbehebung und 8.2 Projektdokumentation. Hier waren für Punkt 1 drei Stunden und für Punkt 2 zwei Stunden geplant. Nachdem nur ein kleiner Fehler aufgetreten ist, konnte die Fehlerbehebung in einer Stunde durchgeführt werden. Die Projektdokumentation hat die restlichen zwei Stunden verzehrt, womit diese insgesamt bei vier Stunden liegt.

	Schritt	Soll	Ist	Differenz
1	Analysephase	11	11	0
2	Entwurfsphase	6	6	0
3	Implementierungsphase	32	32	0
4	Qualitätsmanagement	7	5	-2
5	Abnahme- und Deployment-Phase	5	5	0
6	Erstellen der Dokumentation	6	8	+2
7	Projektbewertung	3	3	0
	Gesamt	70h	70h	0h

9.2 Fazit

Im Zuge des Projektes konnte die Autorin wertvolle Erfahrungen der Planung und Durchführung von Projekten sammeln. Dabei wurde besonders deutlich, von welcher großer Bedeutung stetige Kommunikation untereinander und Rücksprachen mit den Fachbereichen für eine erfolgreiche Projektumsetzung sind. Außerdem konnten neue Erkenntnisse in Bezug auf das Einbinden und Nutzen von Frameworks gewonnen werden. Abschließend kann man sagen, dass die Realisierung des Projektes nicht nur einen Mehrwert für die Fachbereiche bietet, sondern auch für die Autorin eine große Bereicherung war.

9.3 Ausblick

Alle Anforderungen konnten realisiert und zur Zufriedenheit der Stakeholder des Projekts umgesetzt werden. Durch dieses Projekt wurde ein Grundstein für die stetige Weiterentwicklung, Verbesserung und Optimierung der SLA Reporting der YouTrack gelegt.

10 Literaturverzeichnis

iXenso AG, Firmenhistorie 2018

<https://ixenso.com/de/Historie>, Firmenhistorie, Version: 2018

YouTrack Standalone 2018.3 Help

<https://www.jetbrains.com/help/youtrack/standalone/YouTrack-Documentation.html>, Resources for Developers, Version: 2018

YouTrack 6.5.x Documentation

<https://confluence.jetbrains.com/display/YTD65/YouTrack+Documentation>,
YouTrack Documentation, Version: 2015

PHP Documentation

<http://php.net/docs.php>, PHP Manual, version: 2018

JQuery API Documentation

<https://jquery.com/>, API Documentation, version: 2018

© Copyright 2018 iXenso AG

Alle Inhalte dieser Unterlagen sind urheberrechtlich geschützt (Copyright). Das Urheberrecht liegt, soweit nicht ausdrücklich anders gekennzeichnet, bei der iXenso AG. Diese Unterlagen und die beiliegende Software dürfen weder ganz noch teilweise ohne die ausdrückliche, schriftliche Genehmigung von der iXenso AG kopiert werden. Die Informationen in diesem Dokument können sich ohne vorherige Ankündigung ändern. Alle Rechte sind vorbehalten.

Für mehr Informationen kontaktieren Sie bitte:

iXenso AG

Munzinger Str. 1

79111 Freiburg

Telefon +49 761 5903240

Fax +49 761 59032440

Barthstr. 26

80339 München

Telefon +49 89 68088360

Fax +49 89 68088389

www.iXenso.com

mail@iXenso.com

Freiburg | München | Hannover | Frauenfeld (CH)

A. Anhang

A.1. Detaillierte Zeitplanung

Schritt	Dauer
Analysephase	11h
Ist-Analyse	2
Soll-Konzept	4
Erstellen des Lastenheftes	3
Wirtschaftlichkeitsanalyse	2
Entwurfsphase	6h
Entwerfen der Schnittstelle zum Ticket-System	3
Erstellen von Mock-Ups	1
Erstellen eines Use Case Diagramm, Klassendiagramm	2
Implementierungsphase	32h
Entwicklungsumgebung einrichten	4
Erstellen der Benutzeroberfläche	11
PHP actions implementieren	13
Erstellen der Excel Datei	4
Qualitätsmanagement	7h
Durchführung von Tests	4
Fehlerbehebung/ Korrekturen	3
Abnahme und Deployment	5h
Abnahme durch Projektleiter	2
Deployment der Anwendung	3
Erstellen der Dokumentation	6h
Erstellen der Projektdokumentation	4
Erstellen der Anwenderdokumentation	2
Projektbewertung	3h
Soll-/Ist-Vergleich, Fazit	2
Ausblick	1
Gesamt	70h

A.2. Reale Dauer der Projektphasen in Tortendiagramm

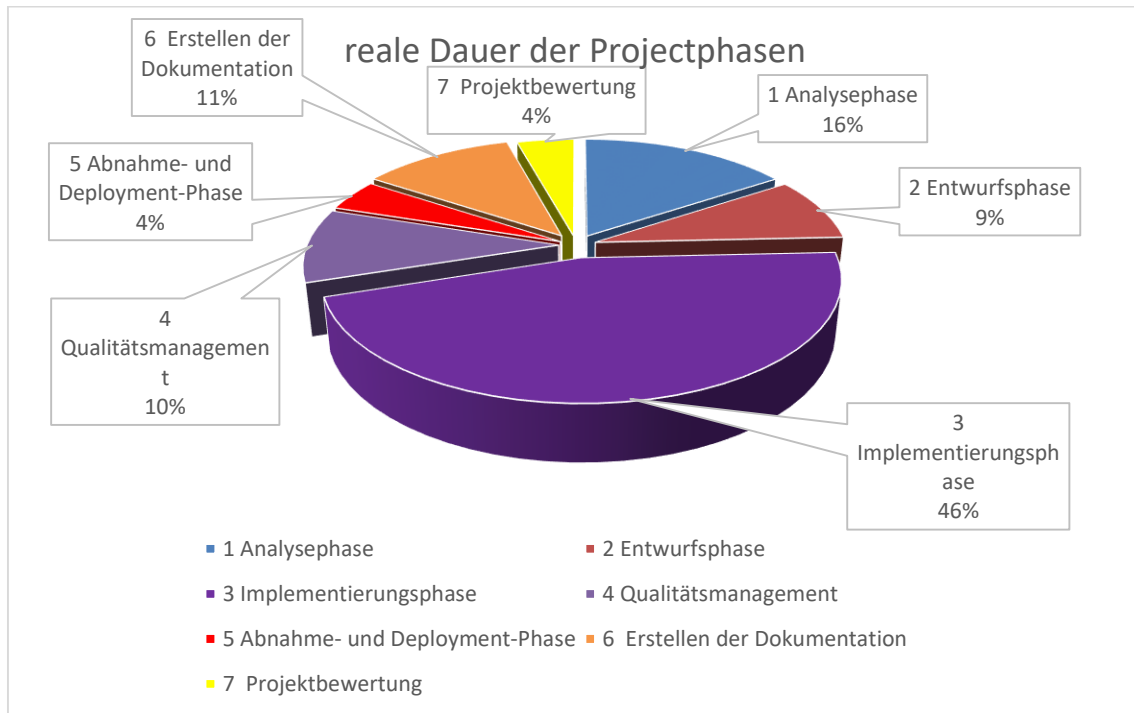


Abbildung 1: Projectphasen in Tortendiagramm

A.3. Gantt Diagramm

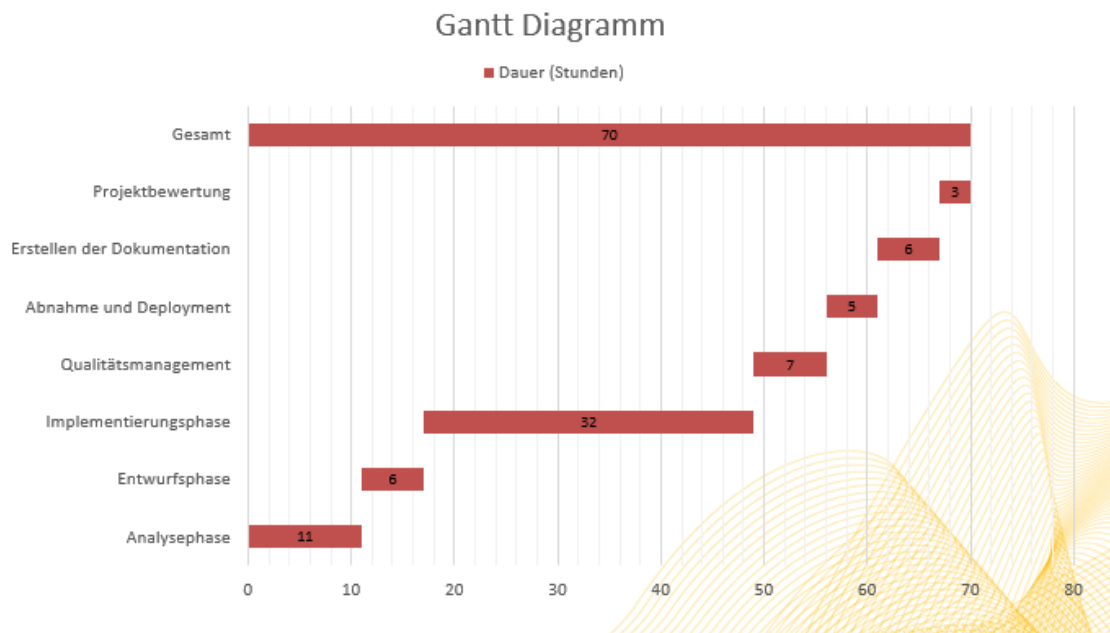


Abbildung 2: Gantt Diagramm

A.4. Verwendete Ressourcen

Hardware

- Büroarbeitsplatz
 - Thin Client: HP ProDesk 600 G3 DM
 - Tastatur
 - Maus
 - Zweiter Monitor
 - Telekommunikationsanlage
- Präsentationsmittel
 - Beamer

Software

- Microsoft Windows 10 Enterprise 2016 LTSC
- PHPEXcel Framework³ 1.8.1 2015
- PHP YouTrack REST von Anton Komarev⁴, release v6.1.0, license MIT
- Holiday Calculation Library for PHP von Mayflower GmbH⁵, license LGPL-3.0⁶
- Zugang auf „issue tracking and project management platform“ (tracker.ixenso.com)
- Microsoft Office Word 2016 – Erstellung der Dokumentationen
- Internet Explorer 7.0 – Web Browser für Darstellung der in PHP entwickelten Websites
- Eclipse Workspace für PHP Entwicklung

Personalplanung

- Ansprechpartner bei allgemeinen Fragen:
 - Frau Yen Nguyen (Auszubildende/Trainee Software Development)
- Ansprechpartner bei Fragen zur Anforderungsanalyse und Projektdurchführung:
 - Herr Matthias van Hout (iXenso Support Team)
 - Herr Udo Kemle (Project Management)
- Ansprechpartner bei technischen Fragen:
 - Herr Thomas Gessner (iXenso Development Team)
- Ansprechpartner bei Fragen zum Datenschutz:
 - Herr Rainer Sarnowski (Datenschutzbeauftragter)

³ Vgl. <https://github.com/PHPOffice/PHPEXcel>

⁴ Vgl. <https://github.com/cybercog/youtrack-rest-php>

⁵ Vgl. <https://github.com/mayflower/libholiday>

⁶ Vgl. <https://github.com/mayflower/libholiday/blob/master/LICENSE>

A.5. Use-Case-Diagramm

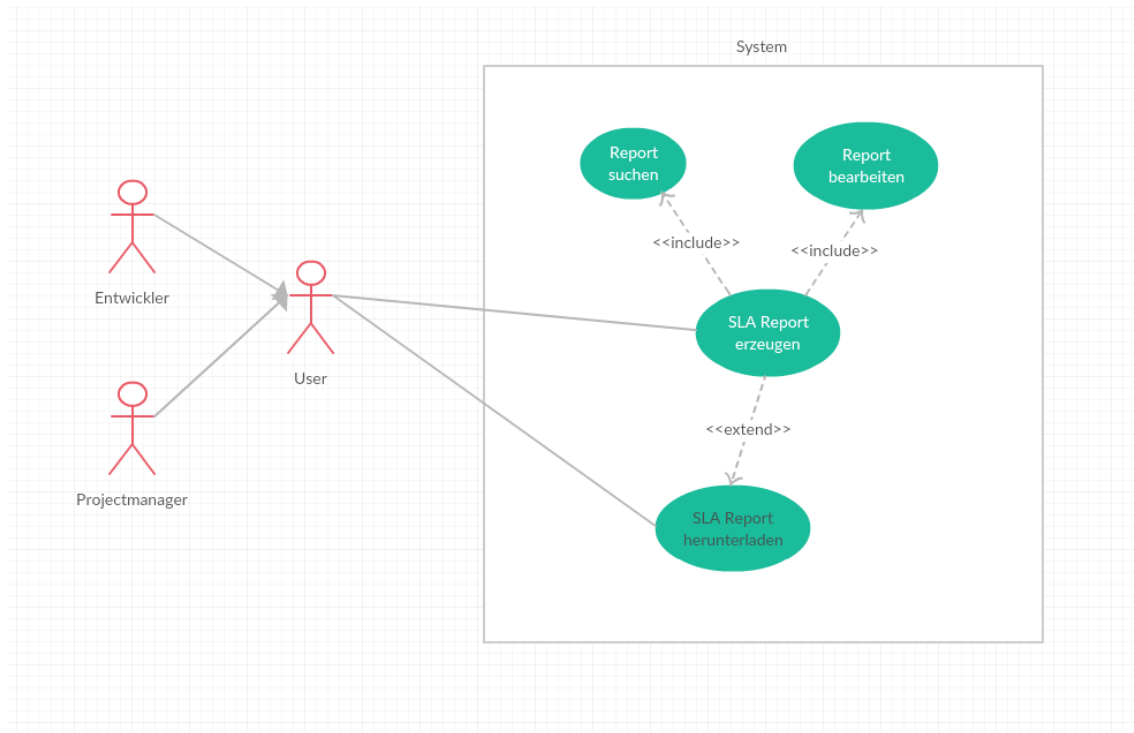


Abbildung 3: Use-Case-Diagramm

A.6. Lastenheft (Auszug)

Im folgenden Lastenheft werden die Anforderungen definiert, welche die zu entwickelnde Anwendung erfüllen muss.

Von der Anwendung müssen folgende Anforderungen erfüllt werden:

- Die Anwendung muss eine Benutzeroberfläche bereitstellen, in welcher die Daten eines bestimmten Projekts, in einem Zeitraum dargestellt und nach Excel exportiert werden können.
- Auf der Benutzeroberfläche gibt es einen Button, der per Klick die angezeigten Daten nach Excel exportiert.
- Reihenfolge der Arbeitsschritte:
 - Auf der Startseite ist das Formular mit den Feldern:
 - Project: Dropdown um Projektname auszuwählen
 - From: Input Zeit *von* wählen
 - To: Input Zeit *bis* wählen
 - Show Report: Button Tabelle Anzeigen klicken oder
 - Download: Button sofort herunterladen klicken
 - Wenn Button Tabelle Anzeigen (Show Report) geklickt wurde, wird eine Seite mit den Daten des gewählten Projekts (Name, Supportzeit/ Erreichbarkeit, Feiertage) und eine Tabelle mit den Tickets im Browser angezeigt.
 - Die angezeigten Ticketdaten sind:
 - Name des Tickets(ticket): Listen allen Ticketnamen eines Projektes
 - Erstellt von (created by): Wer das Ticket erstellt hat
 - Erstellzeitpunkt (create date): Wann das Ticket erstellt wurde
 - Reaktionszeit (time to respond): Die Reaktionszeit beginnt grundsätzlich mit der Erstellung des Tickets in YouTrack bis zur erste Reaktion.
 - Lösungszeit (time to resolve): Von der Erstellung bis zum Status (State) „closed“ oder „verified“, was als erstes passiert.
 - Aktueller Status (current status): Aktueller Status des Tickets
 - Priorität (priority): Priorität des Tickets
 - Die Tabelle soll nach Spalten sortierbar sein.
 - Tickets, welche die Reaktionszeit überschritten haben, sollen farblich rot markiert werden.
 - Wenn der Button herunterladen(Download) geklickt wird, wird eine Excel Report Datei der angezeigten Daten heruntergeladen.

A.7. Oberflächenentwürfe (Mock-Ups)

SLA Reporting

Project

from

to

Show Report

Download

Abbildung 4: Mock-Up der Start Seite

SLA Reporting

Project name

Business time, Feiertage, ...

Ticket name	Created by	Created date	Time to respond	Time to resolved	Current state

Home page

Download

Abbildung 5: Mock-Up der Tabelle Seite

A.8. Pflichtenheft(Auszug)

In folgendem Auszug aus dem Pflichtenheft wird die geplante Umsetzung der im Lastenheft definierten Anforderungen beschrieben:

- Der Excel-Import der Ticketdaten eines Projektes in einem Zeitraum soll direkt über einen Button innerhalb der Anwendung aufgerufen werden können und mit Hilfe des PHPEXcel umgesetzt werden.
- Das Lesen der Daten aus der API Schnittstelle YouTrack soll ebenfalls mit Hilfe des REST API YouTrack erfolgen. Die Berechnung der benötigten Daten soll anhand von der Berücksichtigung der Arbeitszeit und Feiertage vollzogen werden.
- Die Schnittstelle der zu entwickelnden Anwendung soll durch einen Webservice realisiert werden. Dieser Webservice soll anhand der Parameter von übergebenen Ticketnamen das Datum ermitteln und zurückliefern, welche für die Berechnung der Übertragungswerte relevant sind. Außerdem soll er die aus der API Schnittstelle ermittelten Werte abgleichen sowie den Status des zugehörigen Tickets setzen.
- Das objektorientierte Programmiermodell soll mit Hilfe des Model-View-Controller Modell umgesetzt werden.

A.9. Komponentendiagramm

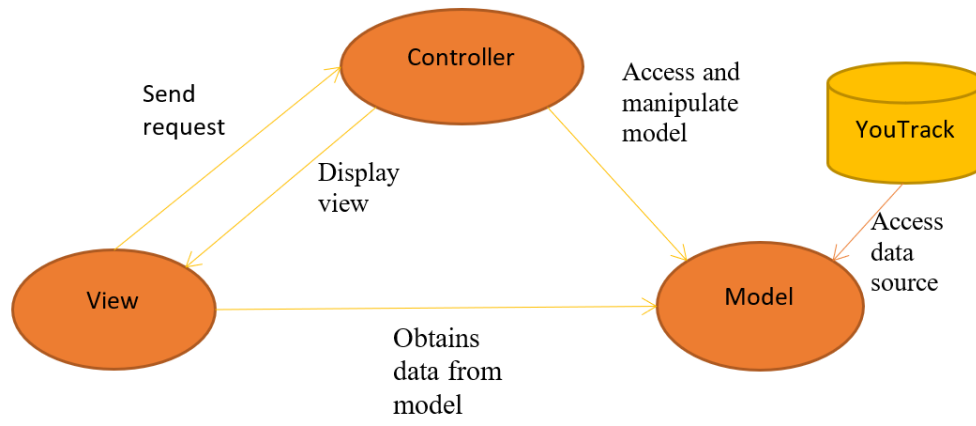


Abbildung 6: Komponentendiagramm

A.10. Screenshots der Start Seite

SLA Reporting

der internen issue tracking and project management platform YouTrack

[Home](#)
[Logout](#)

Project

IXBY
⌵

Choose a project to export SLA-report.

from

07/25/2018

to

10/25/2018

Service time from

08:00

Service time to

17:00

Show Report

download

Abbildung 7: Screenshot der Start Seite

from

07/25/2018

July 2018

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

to

10/25/2018

Service time to

17:00

Abbildung 8: Datum Felder mit Standard Wert und Datepicker

A.11. Screenshots der Tabellenseite

SLA Reporting

der internen issue tracking and project management platform YouTrack

[Home](#) [Logout](#)

IXBY

Business Hours: Describes the time period declared as the most critical to your business and configured while setting up your SLA. The availability SLA will then be applicable for the time frame defined within the Business Hours.

This time period is in Service Time from Monday to Friday. This time frame is applicable everyday except [holidays](#)

Ticket name	Created by	Created date	Time to respond	Time to resolved	Current state	Priority
IXBY-3	Yen Nguyen	08.10.2018	0:16	77:48	Closed	Normal
IXBY-4	Yen Nguyen	08.10.2018	5:29	77:36	Closed	Major
IXBY-2	Yen Nguyen	08.10.2018	0	0	Open	Normal
IXBY-6	Yen Nguyen	16.10.2018	0	0	Open	Normal
IXBY-7	Yen Nguyen	16.10.2018	0	0	Open	Normal
IXBY-8	Yen Nguyen	16.10.2018	0	0	Open	Normal
IXBY-5	Yen Nguyen	08.10.2018	0	0	To be discussed	Normal

[Start Site](#)
[Download](#)

Abbildung 9: Screenshot der Tabellenseite

A.12. Listings der Cookie Authorization

```
use Cog\YouTrack\Rest;

$client = null;

function get_cookie_authorization()
{
    $apiBaseUri = 'https://tracker.ixenso.com';
    if (isset($_SESSION['user'])) {
        $apiUsername = $_SESSION['user'];
        $apiPassword = $_SESSION['password'];

        $psrHttpClient = new \GuzzleHttp\Client([
            'base_uri' => $apiBaseUri,
            'debug' => true
        ]);

        // Instantiate YouTrack API HTTP Client
        $httpClient = new Rest\HttpClient\GuzzleHttpClient($psrHttpClient);

        // Instantiate YouTrack API Cookie Authenticator
        $authenticator = new Rest\Authenticator\CookieAuthenticator($apiUsername, $apiPassword);

        // Instantiate YouTrack API Cookie Authorizer
        $authorizer = new Rest\Authorizer\CookieAuthorizer($authenticator);

        // Instantiate YouTrack API Client
        $client = new Rest\Client\YouTrackClient($httpClient, $authorizer);
        return $client;
    }
}
```

Listing 1: Cookie Authorization

A.13. Listing – Class Controller

```
namespace SLA_Reporting;
use Cog\YouTrack\Rest\Client\YouTrackClient;

abstract class Controller
{
    protected $client_obj;
    protected $view = 'views/';

    /**
     *
     * @param YouTrackClient $client_obj
     */
    function __construct(YouTrackClient $client_obj) {
        $this->client_obj = $client_obj;
    }

    public function view($action) {
        include_once $this->view.'head.php';
        if($action == 'index'){
            include_once $this->view .'login.php';
        }
        include_once $this->view . 'foot.php';
    }
}
```

Listing 2: Class Controller

A.14. Listing - Class Bavaria_Holidays

```
use Holiday\Bavaria;

class Bavaria_Holidays extends Bavaria
{
/**
 *
 * @param string $year
 * @return []
 */
    public function get_holidays_bavaria($year)[]

    public function get_holidays_array($year) {
        $y= $this->get_holidays_bavaria($year);
        $res = [];
        for($i = 0; $i < count($y); $i++) {
            $arr = get_object_vars($y[$i]);
            $res[$i]['date'] = $arr["date"];
            $res[$i]['name'] = $arr["name"];
        }
        return $res;
    }
}
```

Listing 3: Klasse Bavaria_Holidays

A.15. Listing – Index.php

```
use Cog\YouTrack\Rest\Authenticator\CookieAuthenticator;

require_once 'controllers/home_controller.php';
require_once 'function.php';
require_once 'controllers/cookie-authorization.php';

session_start();

$view = "views/";

$client_obj = get_cookie_authorization();

$con = new Home_Controller($client_obj);

if (! empty($_POST)) {
    if (count($_POST) > 0 && array_key_exists('data', $_POST)) {
        $report_data = explode(',', $_POST['data']);
    }
    $con->get_tickets($report_data);
}

$path_info = $_SERVER["PATH_INFO"];
view_html_content($path_info);
```

Listing 4: File index.php

A.16. Listings – Declaration und Navigation der HLML Seite

```

<!DOCTYPE html>
<html lang="en">
⊕ <head>[]

<body>
⊕ <div class="jumbotron titelline text-center">[]

    <nav class="navbar navbar-light navbar-expand-lg navbar-light"
        style="background-color: #e9ecef;">
    ⊖ <div class="container">
        <button class="navbar-toggler" type="button" data-toggle="collapse"
            data-target="#navbarTogglerDemo03"
            aria-controls="navbarTogglerDemo03" aria-expanded="false"
            aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        ⊖ <div class="collapse navbar-collapse" id="navhome"
            style="margin-left: 200px;">
            ⊖ <ul class="navbar-nav mr-auto mt-2 mt-lg-0">

                <?php
                if (isset($_SESSION['user']) && $_SESSION['user'] !== false) {
                    show_home_li();
                    show_logout_li();
                }
                ?>

            </ul>
        </div>
    </div>
</nav>

<div class="container">

    <?php

⊕ function show_home_li()[]
⊕ function show_logout_li()[]
    ?>
  
```

Listing 5: Declaration und Navigation der HLML Seite

A.17. Listing – Methode „excel_export“ in Class PHPExcel_Download

```
function excel_export(array $tickets, array $column_name) {
    $this->objPHPExcel = new PHPExcel();
    $this->objPHPExcel->getProperties()
    ->setCreator("Yen Nguyen")
    ->setLastModifiedBy("Temporal")
    ->setTitle("SLA Reporting")
    ->setSubject("Template excel")
    ->setDescription("Template excel SLA Reporting");
    $this->objPHPExcel->setActiveSheetIndex(0);
    $letterarr = range('A', 'Z');
    $key = $this->get_key();
    for ($i = 0; $i < count($tickets); $i++) {
        for ($ii = 0; $ii < count($column_name); $ii++) {
            $this->objPHPExcel->getActiveSheet()
            ->getColumnDimension($letterarr[$i])
            ->setWidth(20);
            $this->objPHPExcel->getActiveSheet()->SetCellValue(
                $letterarr[$ii].($i+2), $tickets[$i][$key[$ii]]
            );
        }
    }

    $this->write_column_name($column_name);
    //Writertype Excel5
    $writer = PHPExcel_IOFactory::createWriter($this->objPHPExcel, 'Excel5');

    $time_local = $this->get_time_local();

    $filename = "SLA_Reporting_Youtrack_" . $time_local . '.xls';
    //delete file when the same filename exists
    if (file_exists($filename)) {
        unlink($filename);
    }
    $this->set_header($filename);
    $writer->save('php://output');
    echo $writer;
}
```

Listing 6: Methode "excel_export"

A.18. Listing – Class Business_Datetime

```

class Business_Datetime
{
    public $business_timestamp;

    public $holiday;

    public $holiday_name;

    public $typ;

    public $datetime;

    public $fromdate; // "m/d/Y 00:00"

    public $todate;

    private $days;

    private $timezone;

    public $date;

    public $time;

    public $business_time = [];

    function __construct(){}

    * @param \DateTime $fromdatetime[]
    function get_business_time_duration(\DateTime $from, \DateTime $to){}

    * @param DateTime $start[]
    function business_duration_caculation(DateTime $from,DateTime $to) {}

    function get_time_string($array){}

    * @param [] $array[]
    function get_holidays_obj($array){}

    * @param DateTime $datetime[]
    function is_business_time(DateTime $datetime){}

    * @param string $fromdatetime[]
    function get_time_difference_business($fromdatetime, $todatetime){}

    function count_business_day($fromdatetime, $todatetime){}

    function get_next_business_beginning(){

    * @param DateTime $datetime[]
    function get_time_type(DateTime $datetime){}

    * @param string $date[]
    function get_the_next_day($date){}

    * @param Datetime $date1[]
    function is_the_same_date(Datetime $datetime1, Datetime $datetime2){}

}
    
```

Listing 7: Class Business_Datetime

A.19. Listings – Event-Handler der API-Schnittstelle (Auszug von start_form.js)

```
$(document).ready(function () {
    $("#download").click(function(event){
        var x = $("option[selected='selected']").val();
        if(typeof x !== 'undefined') {
            redirect_get('download');} else { input_error(); }
    })
    $("#show_report").click(function(event){
        var x = $("option[selected='selected']").val();
        if(typeof x !== 'undefined') {
            redirect_get('table');} else { input_error(); }
    })
    function redirect_get(site) {
        reset_input_error();
        var fromdate = $("input#fromdate").val();
        var todate = $("input#todate").val();
        var servicefrom = $("input#servicefrom").val();
        var serviceto = $("input#serviceto").val();
        var x = $("option[selected='selected']").val();
        if(servicefrom > serviceto) {

            $("#message").fadeOut(0, function () {
                $("#div.servicefrom").css('color', 'red');
                $("#div.serviceto").css('color', 'red');
                $("#input#serviceto").css('border-color', 'red');
                $("#input#servicefrom").css('border-color', 'red');
                $(this).html("<div class='alert alert-danger'>Service time from muss be earlier than Service time to. </div>").fadeIn();
            })
        } else {
            var url = site+"?report="+x+"&fromdate="+fromdate+"&todate="+todate+"&servicefrom="+servicefrom+"&serviceto="+serviceto;
            window.location = url;
        }
    }

    var today = new Date();
    var dd = today.getDate();
    var mm = today.getMonth()+1; //January is 0!
    var yyyy = today.getFullYear();

    if(dd<10) { dd = '0'+dd }
    if(mm<10) { mm = '0'+mm }
    today = dd + '.' + mm + '.' + yyyy;
    $("#fromdate").datepicker({
        //   yearRange: "2014:2015",
        //   maxDate: "+0y",
        //   minDate: todate,
        //   dateFormat: 'dd.mm.yyy',
    });
});
```

Listing 8: Event-Handler JavaScript der API Schnittstelle

A.20. Listings - Auszug von Klasse Home_Controller

```
class Home_Controller extends Controller
{

    private $tickets_data_from_youtrack;
    private $edited_tickets_array;//edited array

    private $project_name;
    private $from_date;
    private $to_date;

    private $from_timestamp;
    private $to_timestamp;

    private $project_model_object;
    private $ticket_model_object;

    function __construct($client_obj)
    {
        $this->client_obj = $client_obj;
        $this->project_model_object = new Project_Model();
    }

    function get_projects()
    {
        // https://tracker.ixenso.com/rest/admin/project
        $response = $this->client_obj->get('/admin/project');
        $projects = $response->toArray();
        foreach ($projects as $project) {
            $res[] = $project['id'];
        }
        return $res;
    }

    function tickets_sort($array, $on, $order=SORT_ASC)[]

    function action_index() {
        $this->view('index');
    }

}
```

Listing 9: Klasse Home_Controller

A.21. Listing – Class Ticket_Model

```

class Ticket_Model extends Model
{
    public $ticket_name;
    public $creator;
    public $created_timestamp;
    public $created_datetime;
    public $time_to_respond;
    public $respond_timestamp;
    public $time_to_resolved;
    public $resolved_datetime;
    public $state;
    public $comment;
    public $client_obj;
    public $business;
    public $priority;
    private $ticket_changes;
    private $first_comment_time = "0";
    private $change_data;
    private $first_state_change = "0";
    + function __construct($client_obj){}
    + public function get_ticket(){}
    + static function get_ticket_data(Ticket_Model $ticket_model){}
    + public function set_ticket_data_from_id($ticket_id){}
    + private function get_ticket_data_from_changes($ticket_id){}
    + private function set_ticket_data(Ticket_Model $ticket_model){}
    + function caculate_time_to_respond(){}
    + private function caculate_time_to_resolved(){}
    + function milisecond_to_time($time){}
}
    
```

Listing 10: Klasse Ticket_Model

A.22. Listing – Class Project_Model

```

class Project_Model extends Model{
    private $project_name;
    private $creator;
    private $created_time;
    private $ticket_array;

    private $tickets_data_from_youtrack;

    public function __construct() {

    }

    function get_project_data() {
        $res['project_name'] = $this->project_name;
        $res['creator'] = $this->creator;
        $res['created_time'] = $this->created_time;
        $res['tickets'] = $this->ticket_array;

        return $res;
    }

    * @param YouTrackClient $client_obj[]
    public function set_ticket_array(YouTrackClient $client_obj, array $ticket_array) {}

    * @param DateTime $fromdate[]
    function get_tickets_array_by_date(DateTime $fromdate, DateTime $todate) {}

    function milisecond_to_time($time) {}

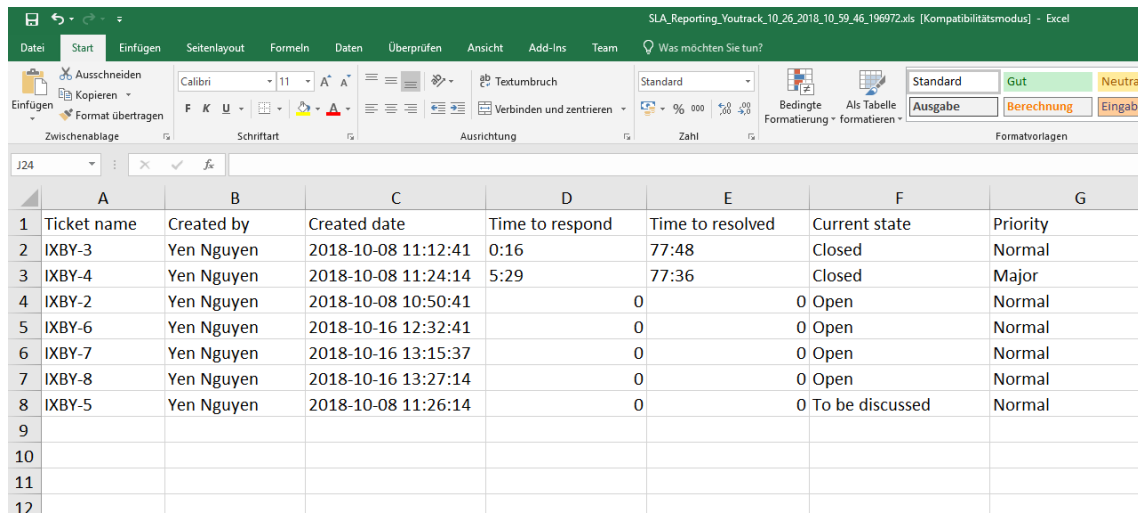
    function get_ticket_models_from_project() {}
}

```

Listing 11: Klasse Projekt_Model

A.23. Anwenderdokumentation

Im folgenden Auszug aus der Anwenderdokumentation werden die einzelnen Spalten der exportierten Exceldatei genau beschrieben.



	A	B	C	D	E	F	G
1	Ticket name	Created by	Created date	Time to respond	Time to resolved	Current state	Priority
2	IXBY-3	Yen Nguyen	2018-10-08 11:12:41	0:16	77:48	Closed	Normal
3	IXBY-4	Yen Nguyen	2018-10-08 11:24:14	5:29	77:36	Closed	Major
4	IXBY-2	Yen Nguyen	2018-10-08 10:50:41	0	0	Open	Normal
5	IXBY-6	Yen Nguyen	2018-10-16 12:32:41	0	0	Open	Normal
6	IXBY-7	Yen Nguyen	2018-10-16 13:15:37	0	0	Open	Normal
7	IXBY-8	Yen Nguyen	2018-10-16 13:27:14	0	0	Open	Normal
8	IXBY-5	Yen Nguyen	2018-10-08 11:26:14	0	0	To be discussed	Normal
9							
10							
11							
12							

Abbildung 10: exportierte Excel Report Datei

- Ticket name: Name des Tickets über ein Projekt, welches in dem ausgewählten Zeitrahmen erzeugt wurde.
- Created by: Vor- und Zuname des Users, der das Ticket erstellt hat.
- created date: Datum und Uhrzeit, wann das Ticket erstellt wurde.
- time to respond: Die Reaktionszeit beginnt mit der Erstellung des Tickets in YouTrack bis zur ersten Reaktion (hier wird beim ersten Mal ein Kommentar eingefügt oder wann der Status zum ersten Mal geändert wurde). Die Zeit wird so dargestellt: Tage:Stunde:Minute. Wenn das Ticket noch nicht abgeschlossen ist, steht hier nur eine Zahl 0.
- time to resolved: Zeit von der Erstellung bis Status (State) „closed“ oder „verified“, was als erstes passiert. Die Zeit hier ist so dargestellt: Tage:Stunde:Minute. Wenn das Ticket noch nicht abgeschlossen ist, steht hier nur eine Zahl 0.
- current status: Aktueller Status des Tickets, welcher als Letztes im YouTrack gespeichert wurde.
- Priority: Priorität des Tickets

A.24. Get_business_time_duration Beschreibung

get_business_time_duration: Die Dauer der Arbeitszeit zwischen 2
Datumswerten

```
public String get_business_time_duration (DateTime $datetime1, DateTime  
$datetime2)
```

Parameter Liste:

\$datetime1: das erste Datum zu vergleichen

\$datetime2: das zweite Datum zu vergleichen

Rückgabewerte:

Ein String präsentiert die Dauer zwischen 2 Datumsangaben in Stunden und
Minuten mit format „Stunden:Minuten“.

A.25. Klassendiagramm

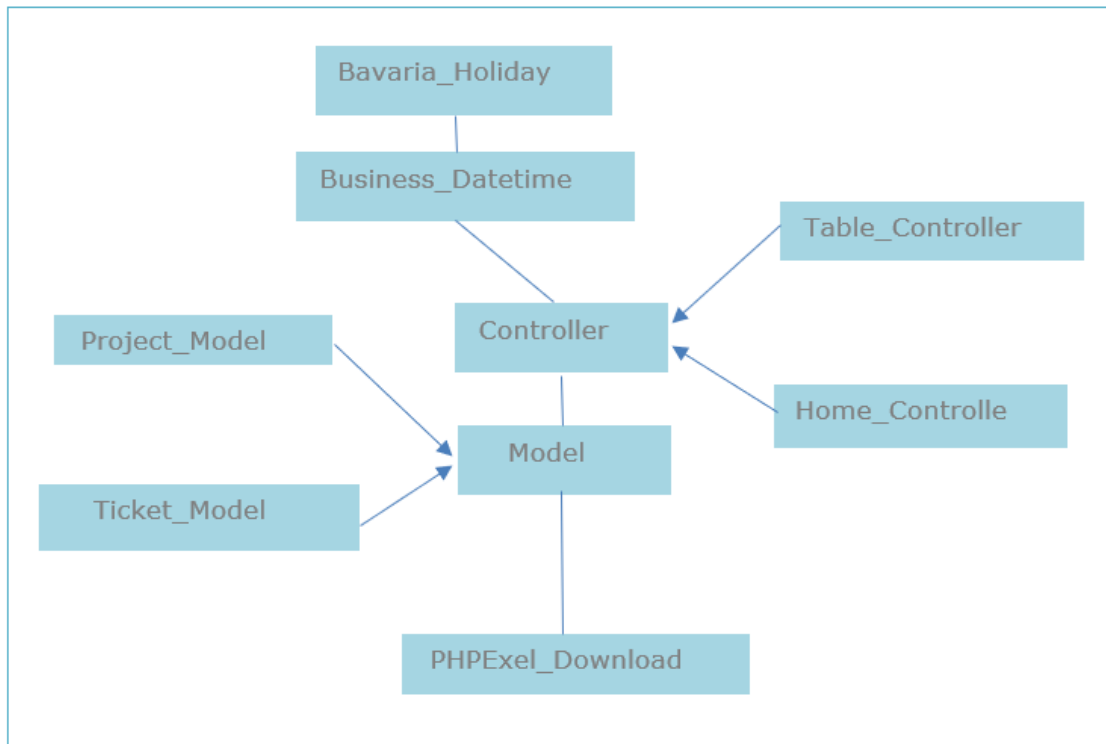


Abbildung 11: Klassendiagramm