**X.Projektbeschreibung**

**X.x Projektziel:**

Es soll eine Android-App entwickelt werden, welche zur Erfassung schulinterner Hardware (z.B. Rechner, Bildschirm,…) mit Inventarnummern genutzt werden kann. Mit der App soll ein Barcode (Inventarnummer) gescannt und weitere Daten zum Gerät erfasst werden. Diese Daten sollen anschließend in eine csv-Datei exportiert werden.

**X.x Projektbegründung:**

Die Schule „Ferdinand-Braun-Schule“ benötigt zur besseren Verwaltung der Schulhardware eine App.  
Ansprechpartner zur Entwicklung der App ist Herr Reuter. Die App soll eine schnelle und effiziente Erfassung von den Geräten ermöglichen. Aktuell gibt es nur eine Liste für die Gesamtheit aller Geräte und keine explizite Zuordnung zu den Klassenräumen.

**X.x Projektschnittstellen:**

Die App soll für Android ab der Version 5 (Lollipop, API 21) verfügbar sein. Die App macht von einem externen Barcodescanner Gebrauch, der installiert sein muss. Dieser benötigt die “Google-Standard-Bibliothek“. Das Format ist Comma-separated values (CSV). Hierbei werden die einzelnen Spalten durch „ ; “ getrennt.

**x.x Ressourcenplanungen**

Bei der Ressourcenplanung wurde drauf geachtet, dass die die Software zur freien Verfügung steht, oder die Firma die Lizenzrechte hat, um im möglichst kleinen Budget zu bleiben. Da dies ein sehr kleines und simples Projekt ist, entschied man sich die Azubis alles eigenständig zu machen. Somit kommt man auf geringe Kosten und der Azubi hat hierbei einen enormen Lernfortschritt gemacht.

**X.x Entwicklungsprozesse**

Bevor mit der Realisierung des Projektes begonnen werden konnte, musste ein geeigneter Entwicklungsprozess gewählt werden, welcher die Vorgehensweise definiert, nach der die Umsetzung erfolgen soll. Die beiden Entwickler haben sich unter Abstimmung mit dem Auftraggeber für einen agilen Entwicklungsprozess entschieden. Dieser ermöglicht es, flexibler und transparenter mit dem Kunden zusammenzuarbeiten, da regelmäßig Rücksprache gehalten wird und somit schneller auf kurzfristige Wünsche und Änderungen eingegangen werden kann. Die agile Softwareentwicklung zeichnet sich zudem durch einen möglichst geringen bürokratischen Aufwand aus, wodurch ein schneller Einsatz der entwickelten Systeme möglich wird. Aufgrund der Entscheidung für diesen Entwicklungsprozess sind die geplanten Zeiten für Projektplanung, Analysephase und Entwurfsphase in Abschnitt X.X relativ niedrig angesetzt worden. Die wichtigste Phase ist die der Implementierung, da viele Besonderheiten erst in diesem Zeitraum geklärt werden können und auch das Testen in dieser geschieht. Durch regelmäßige Kommunikation mit dem Fachbereich und häufiges Feedback können die Entwickler deutlich besser den gegebenen Anforderungen gerecht werden. Der ständige Kontakt hat weiterhin den Vorteil, dass sich der Auftraggeber bereits während des Entwicklungsvorgangs mit dem entstehenden Produkt vertraut machen kann. Dadurch kann Zeit bei der abschließenden Abnahme und Einführung gespart werden, weshalb auch für diese Phase nur wenig Zeit vorgesehen wurde.

**x Analyse**

**x.1 Ist-Analyse**

Momentan sind die Geräte mit Inventarnummer festgehalten, aber dort ist nicht der Raum angegeben. Somit muss man die Geräte Raum für Raum durchsuchen. Dies stellt einen sehr großen Zeitaufwand dar.

**x.x Soll-Analyse**

Die Ferdinand-Braun-Schule wünscht sich eine App, bei der man die Inventarnummer von Geräten(PC,Monitor, Beamer, Drucker) erfassen kann. Hierzu soll der Benutzer ggf. Bemerkungen tätigen. Der Barcodescanner muss nicht integriert werden, sondern kann auch über einen externen Barcodescanner aufgerufen werden. Ist ein Raum vollständig erfasst, so soll eine CSV-Datei erstellt werden. Diese kann dann über USB auf den PC übertragen werden. Sollte das Projekt stark unter den Kosten sein, dürfen noch weitere Features wie ein übersichtliches Design, anzeigen von Räumen, sowie das Löschen von Räumen in der App programmiert werden.

Darauf achtet die Ferdinand-Braun-Schule nach Wichtigkeit:

|  |
| --- |
| -Scannen von Barcodes |
| -Pro Raum CSV-Datei erstellen |
| -Datei über USB auf PC |
| -Anzeige von Räumen |

**X.x Zeitplanung**

Unsere Zeitplanung setzte sich aus der Planungsphase, der Analyse, der Entwurf dann die Implementierung und Programmierung. Im Anschluss folgten Qualitätskontrollen (Tests). Zum Schluss folgte die Abnahmephase mit Schulung. Die Projektdokumentation wurde zwischen den einzelnen Phasen ergänzt und zum Schluss fertig bearbeitet.

Eine Zeitschätzung erfolgt nun in tabellarischer Form:

|  |  |
| --- | --- |
| **Projektphase** | **Schätzung in Stunden** |
| Planungsphase | 6,5 Stunden |
| Analysephase | 4,5 Stunden |
| Entwurfsphase | 5 Stunden |
| Implementierung/Programmierung | 34,5 Stunden |
| Qualitätskontrolle | 5 Stunden |
| Abnahme | 2 Stunden |
| Projektdokumentation | 12,5 Stunden |
| **Gesamt:** | **70 Stunden** |

**4 Entwurf**

**4.1 Ablaufplan**

Damit man eine grobe Vorstellung des Programms hat, wie es ablaufen soll (Benutzereingaben) findet man im Anhang einen Ablaufplan, der die Eingaben der Benutzer verwaltet. Der Ablaufplandesgin wurde schnellstmöglich für den Kunden entwickelt, um weitere Missverständnisse aufzudecken. Hierbei wird nur der Ablauf als solches dargestellt. Der Ablaufplan dient nur zur groben Vorstellung wie das Spiel Ablaufen könnte, bzw. sein sollte. Der Ablaufplan findet sich im Anhang unter dem Punkt „A4.1“.

**X.1 Programmierung**

**X.1.2 Layout**

**x.1.2.1 Allgemeines**

Das Layout hat als Hintergrundfarbe ist Mintgrün. Die Buttons haben alle einen lachsfarbenen Hintergrund. Jede Schrift ist weiß. Außerdem gibt es auf jeder Activty in der rechten oberen Ecke ein Zurück-Button (Hauptmenü ausgeschlossen). Jede Activty besitzt in der Titlebar einer kurzen Beschreibung, in welcher Activity wir uns befinden.

**x.1.2.2 Hauptmenü**

Ich Hauptmenü befinden sich 2 Buttons, sowie ein Imagebutton. Der Imagebutton öffnet im Browser die FBS-Homepage. Der Button mit dem Text „Raum erfassen“ öffnet einen kleinen Dialog, der Titel ist „Raum erfassen“ darunter steht „Bitte Raumnamen eingeben“. Somit ist dort ein EditText zu sehen, in der man die Raumnummer eingeben muss. Ist die Raumnummer leer, so wird man mit einem Toast darauf hingewiesen, dass die Raumnummer leer ist, und man zum Fortsetzen eine Raumnummer eintragen muss.

Dann gibt es noch ein Button mit dem Text „Räume Anzeigen“. Dieser öffnet eine neue Acticty und zwar Anzeige Räume. Sind jedoch keine Daten vorhanden, so wird ein Toast ausgegeben. (siehe Anhang A.x.1)

**x.1.2.3 Scannen**

In dieser Activity sehen wir links ein TextView mit Raumnamen, daneben steht der Raum, welchen wir in dem Dialog eingetragen haben. Darunter ist Gerätetyp, rechts davon ist ein Spinner, klickt man auf diesen drauf, so öffnet sich dieser und man kann sich zwischen den einzelnen Gerättypen unterscheiden. Standardmäßig ist dieser PC. Dann sieht man Inventarnummer und daneben ein TextEdit mit „Letzter Scan“. Hier kann man entweder das Feld manuell befüllen, oder man lässt dies mithilfe des Buttons „Scannen“ füllen. Dieser öffnet dann einen Barcodescanner, oder fragt ob man einen Installieren will. Ist die Inventarnummer erfolgreich eingetragen, so kann man nun eine Bemerkung im „EditText“ eintragen. Klickt man nun auf den Button „Gerät hinzufügen“ so wird das Gerät hinzugefügt, die Bemerkung und die Inventarnummer werden wieder auf „“ gesetzt und unter dem Button erscheint nun die Anzahl der erfassten Geräte. Ist die Inventarnummer jedoch vorher „“ dann wird ein Toast erscheinen mit folgender Nachricht „“Inventarnummer ist leer!“ Klickt man nun auf den Button „Raumerfassung abschließen“, wird die CSV-Datei erzeugt, und man ist wieder im Hauptmenü.

Drückt man die Zurück-Taste, erscheint eine Meldung, ob man die „Erfassung abbrechen“ will. Hierbei wird unterschieden, ob sich Daten im Zwischenspeicher befinde, oder ob kein Scan stattgefunden hat. (siehe Anhang A.x.1)

**x.1.2.4 Anzeige Räume**

Hier befindet sich ein Listview mit den Angezeigten Räumen. Rechts oben befinden sich 3 Punkte, mit diesen kann man alle Dateien löschen. Hierbei wird wieder ein Dialog abgefragt, ob man wirklich alle Daten löschen will. Zudem kann man auch einzelne Räume löschen, dies funktioniert mit einem länger gedrückt halten des Raumes. Dann öffnet sich ein kleines Popup-Menü, bei der man die Option löschen hat. Drück man nur kurz auf einen Raum, so öffnet sich die Activty „Raumdetails“. Sollten sich Daten während der App ändern(ersetzen

der Datei, während Handy am PC offen ist) kann man bequem über das „Swipe-to-Refresh“-Layout die Daten aktualisieren. (siehe Anhang A.x.1)

**x.1.2.5 Anzeige Raumdetails**

In den Raumdetails sieht man eine Tabelle, hier kann man wieder über das „Swipe-to-Refresh“-Layout die Daten aktualisieren. Über den Zurückbutton kommt man in das vorherige Layout.

**x. Qualitätskontrolle**

**x.1 Genauer Programmablauf**

Der genaue Programmablauf findet sich im Anhang wieder, dieser wurde mit einer Android Studio Extension automatisch erzeugt. Wir haben kleine Änderungen vorgenommen und bei unübersichtlichen stellen Kommentare verfasst. *Bild im Anhang A6.1*

**x.2 Beschreibung der Test**

Da wir das System des agilen Protoyping verwendet habe, gab es keine automatischen Tests. Diese wurden alle manuell angesteuert. Bugs wurden schnell gefixt. Es wurde Schrittweiße geschaut ob die jeweilige Komponente funktioniert, dabei wurde schon während der Programmierung Tests gemacht, ob die bis dahin vorgestellten Anforderungen erfüllt sind. Nachdem wir der Meinung waren, dass die Programm fertig ist, wurden ausführlichere Test durchgeführt.

**x.3 Manuelle Test**

Die ausführlichen Tests liefen soweit erfolgreich ab, kleinere Bugs, die man erst nach der Programmierung sehen konnte, wurden mit wenig Zeitaufwand behoben, dadurch bekommt der Kunde eine bugfreie App. Bei dem Test wurden alle möglichen Varianten durchgetestet und die wurden mit ein paar Beispieldaten gefüllt, um zu schauen, wie es sich in der Praxis schlägt, oder es Verbesserungsvorschläge bekommen könnte

**7 Abnahme**

**X.1 Installation**

**X.2 Entwicklerdokumentation**

Es gibt keine Entwicklerdokumentation, das Programm ist im Code gut kommentiert, daraus lassen sich die nötigen Kenntnisse schließen, um das Programm als solches zu erweitern. Zudem findet man im Anhang *„Punkt A5.2.1“* ein Klassendiagramm und unter *„Punkt A6.1“* ein Ablaufdiagramm und ei.

**X.3 Benutzerdokumentation**

Für die Entwicklerdokumentation schauen sie bitte im Anhang „*A7.2“,* dort ist das Programm ausführlicher beschrieben

1. **Anhang**

**A.x Benutzeranleitung**

**A.x.1 Hauptmenü:**

Öffnet der Benutzer die App, ist man im Hauptmenü, findet man drei Buttons. Der oberste Button (FBS-Logo) öffnet die FBS-Homepage(URL: <https://www.ferdinand-braun-Schule.de/blog/category/news/>). Der zweite Button öffnet mit der Schrift „Raum erfassen“. Betätigt man diesen, öffnet sich eine Dialogbox mit dem Text „Raum erfassen“. Klickt man in den Edittext, öffnet sich die Tastatur und man kann den Raumnamen eingeben. Der letzte Button ist der Button „Räume anzeigen“ Hierbei wird eine weitere Activty geöffnet.

Screenshot: siehe A..x

**A.x.2 Erfassung – Raum:**

Trägt man erfolgreich den Raumnamen ein, so kommt man zur Erfassungsmaske „Erfassung Raum“. Als erstes sieht man den Raum, darunter ist der Gerätetype zu sehen. Klickt man den Spinner, an, so öffnet sich dieser und man kann zwischen den Geräten „PC“,“Monitor“,“Drucker“,“Beamer“ auswählen. Standardmäßig ist der „PC“ ausgewählt. Hat man den Gerätetyp bestimmt, so kann man die Inventarnummer erfassen. Hierbei hat man zwei Möglichkeiten. Möglichkeit Nummer eins , man klickt in das Edittext „Letzter Scan“ und trägt die Inventarnummer ein. Die andere Variante ist, man klickt auf das Feld Scannen. Hierbei öffnet sich eine Barcodescanner-App, sofern diese vorhanden ist. Sollte sie nicht vorhanden sein, so schlägt er eine Barcodescanner-App vor und man kommt zum Play-Store. Ist die Inventarnummer eingetragen, so kann man nun eine Bemerkung hinzufügen, diese muss nicht befüllt sein. Ist das Gerät erfolgreich erfasst wurden, drückt man den Button „Gerät Hinzufügen“ . Dabei ist nun eine „TextView“ sichtbar, die anzeigt, wie viele Geräte man erfasst hat. Hat man alle Geräte erfolgreich erfasst, klickt man auf den Button „Raumerfassung abschließen“. Nun ist man wieder im Hauptmenü.

Screenshot: siehe A..x

**A.x.3 Erfasse Räume:**

Hier wird eine Übersicht dargestellt, welche räume man erfasst hat. Die Räume werden sortiert nach Nummer und Alphabet angezeigt. Über die drei Punkte in der rechten oberen Ecke kann man alle erfassten Räume löschen. Hierbei kommt eine Meldung, ob man wirklich alle Räume löschen will. Bestätigt man das mit „Ja“ werden alle Räume gelöscht. Hält man länger gedrückt auf den Raumnamen, kommt ein kleines Pop-Up mit der Funktion löschen, klickt man hier auf „Löschen“ wird der Raum gelöscht. Hierbei erscheint keine Meldung. Klickt man nur kurz auf den Raumnamen so öffnet sich die Maske „Zeige Raumdetails“. Screenshot: siehe A..x

**A.x.4 Raumdetails:**

In den Raumdetails findet sich eine Tabelle wieder. Die Überschriften sind „Gerätetyp“, „Inventarnummer“ und „Notiz“. Darunter werden die erfassten Geräte angezeigt. Bei dem Feld „Notiz“ wird nicht die komplette Bemerkung angezeigt. Die Raumdetails sollen nur einen kurzen Überblick geben über erfasste Geräte.

Screenshot: siehe A..x

**A.x.5 Dateien als CSV-Datei exportieren:**

Als erstes Verbindet man das Handy mit dem PC, danach muss man bei dem Handy „Dateien Übertragen auswählen. Nun ist man im Wurzelverzeichnis. Dort findet man einen Ordner Namens: „FBS“. Dort sind alle CSV-Dateien. Diese können nun in das gewünschte Verzeichnis verschoben werden.